

SISTEMA DE SERVICIOS PREVISIONALES

MIGUEL LEONARDO ORTEGA PEÑUELA

FUNDACIÓN UNIVERSITARIA LOS LIBERTADORES
FACULTAD DE INGENIERÍA
INGENIERÍA DE SISTEMAS
BOGOTÁ
2016

SISTEMA DE SERVICIOS PREVISIONALES

MIGUEL LEONARDO ORTEGA PEÑUELA

Proyecto de Grado realizado para optar por el título en Ingeniería de Sistemas

ING. AUGUSTO JOSE ANGEL MORENO
Director

FUNDACIÓN UNIVERSITARIA LOS LIBERTADORES
FACULTAD DE INGENIERÍA
INGENIERÍA DE SISTEMAS
BOGOTÁ
2016

Nota De Aceptación:

Firma del Presidente del Jurado

Firma del Jurado

Firma del Jurado

Bogotá 11 de Mayo de 2016

A mi esposa Sandra y a mis hermanos Sebastián y Sofía, este logro no es solo mío, este logro es por y para ustedes que han estado durante tanto tiempo apoyándome, siendo el motor de mis decisiones y acompañándome sin importar donde vaya.

AGRADECIMIENTOS

Agradezco a las empresas Imagine SAS y Protección Pensiones y Cesantías, por darme la oportunidad de realizar este proyecto, por permitirme ejercer mi carrera y por confiar en mis capacidades para llevar adelante este proyecto.

Al Ing. Augusto José Ángel, por su orientación y su apoyo con este trabajo, por sus recomendaciones y sobre todo por la paciencia.

CONTENIDO

AGRADECIMIENTOS.....	5
CONTENIDO	6
LISTA DE TABLAS	10
LISTA DE FIGURAS	11
GLOSARIO	12
RESUMEN.....	14
Palabras clave.....	14
ABSTRAC	15
Keywords.....	15
1. INTRODUCCIÓN.....	16
2. JUSTIFICACIÓN.....	17
3. OBJETIVOS.....	18
3.1. Objetivo General.....	18
3.2. Objetivos Específicos.....	18
4. MARCO TEÓRICO	19
4.1. Definición	19
4.1.1. Primera Definición	19
4.1.2. Segunda definición	19
4.2. Ventajas de un workflow	20
4.3. Integración	20
4.4. Introducción a programas Workflow	20
4.5. Workflow en el sector financiero	21
4.5.1. Olivetti	22
4.5.2. Staffware	22
4.5.3. Bruce Appleton, Imaging & Workflow Manage Mercantile Mutual	22
4.6. Otras aplicaciones de los Workflows	23
4.6.1. Workflows Consuetudinarios	23
4.6.2. Woorkflows Manuales	23
4.6.3. Workflows Implícitos.....	23
4.6.4. Workflow Documentales.....	23
4.6.5. Workflow de Aplicaciones.....	24
4.6.6. Workflow ad-hoc.....	24

5. INGENIERÍA DEL PROYECTO	25
5.1. Descripción de la Situación Actual.....	25
5.2. Requerimientos de la Información	26
5.2.1. Requerimientos funcionales	26
5.2.1.1. Asesoría Preliminar	26
5.2.1.2. Carga Masiva de Documentos	26
5.2.1.3. Validación Documental.....	26
5.2.1.4. Radicación de Solicitudes	26
5.2.1.5. Categorización de Solicitudes	27
5.2.1.6. Solución de Inconsistencias	27
5.2.1.7. Definición del Trámite.....	27
5.2.1.8. Reportes de la aplicación	27
5.2.1.9. Ficha General.....	27
5.2.2. Requerimientos no funcionales	28
5.2.2.1. Requerimientos de Sistema Operativo	28
5.2.2.2. Perfilamiento	28
5.2.2.3. Lenguaje de Programación	28
5.2.2.4. Almacenamiento de Datos	28
5.2.2.5. Soporte a la aplicación	28
5.3. Modelamiento del Sistema.....	28
5.3.1. Diagramas de Casos de Uso.....	28
5.3.1.1. Caso 1 – Solicitud Asesoría preliminar.....	28
Figura 1 - Caso de uso 1, solicitud de asesoría preliminar.....	29
5.3.1.2. Caso 2 – Validación de Documentos	30
Figura 2 - Caso de uso 2, validación de documentos.....	30
5.3.1.3. Caso 3 – Validación de Inconsistencias	30
Figura 3 - Caso de uso 3, validación de inconsistencias.....	31
5.3.1.4. Caso 4 – Agendamiento.....	31
Figura 4 - Caso de uso 4, agendamiento.	31
5.3.1.5. Caso 5 – Radicación en firme	32
Figura 5 - Caso de uso 5, radicación en firme.....	32
5.3.1.6. Caso 6 – Validación de Formatos	33
Figura 6 - Caso de uso 6, validación de formatos.	33

5.3.1.7. Caso 7 – Validación de Inconsistencias	33
Figura 7 - Caso de uso 7, validación de inconsistencias.....	33
5.3.1.8. Caso 8 – Normalización de Inconsistencias.....	33
Figura 8 - Caso de uso 8, normalización de inconsistencias.....	34
5.3.1.9. Caso 9 - Trámite de Bono	34
Figura 9 - Caso de uso 9, trámite de bono.....	34
5.3.1.10. Caso 10 – Definición de la solicitud	34
Figura 10 - Caso de uso 10, definición de la solicitud.	35
5.3.2. Script de la base de datos.....	36
5.3.3. Diccionario de Datos	75
5.3.3.1. Listado de tablas	75
Tabla 1 Listado general de tablas del sistema	75
5.3.3.2. Diccionario de Datos	78
Tabla 2 Diccionario de datos	78
5.4. Descripción del Sistema	139
6. EVALUACIÓN ECONÓMICA DEL PROYECTO.....	140
6.1. Riesgo en fase de análisis	140
6.2. Riesgo en fase de diseño	140
6.3. Riesgo en fase de codificación	140
6.4. Riesgo en fase de pruebas	141
6.5. Riesgo en fase de implementación	141
6.6. Riesgo en fase de mantenimiento	141
7. PRESUPUESTO DETALLADO	142
7.1. Costo de infraestructura física	142
Tabla 3 Costo de la infraestructura física del proyecto.....	142
7.2. Costo Total del Proyecto.....	142
Tabla 4 Tabla de costos generales del proyecto	142
8. BENEFICIOS DE IMPLEMENTACIÓN	143
8.1. Operacionales.....	143
8.2. De gestión.....	143
8.3. Estratégicos	143
8.4. De Infraestructura	144
8.5. De Tecnologías de la Información (IT).....	144

9. ALCANCES DEL PROYECTO	145
10. LIMITACIONES DEL PROYECTO.....	146
11. CRONOGRAMA DE ACTIVIDADES.....	147
Tabla 5 Cronograma de Trabajo	147
12. RECOMENDACIONES	148
13. CONCLUSIONES	149
BIBLIOGRAFÍA	150
WEBGRAFÍA	151
Anexos del Documento	152

LISTA DE TABLAS

	pág
Tabla 1. Listado general de tablas del sistema	74
Tabla 2. Diccionario de datos	77
Tabla 3. Costo de la infraestructura física del proyecto	141
Tabla 4. Tabla de costos generales del proyecto	141
Tabla 5. Cronograma de Trabajo	146

LISTA DE FIGURAS

	pág
Figura 1. Caso de uso 1, solicitud de asesoría preliminar.	28
Figura 2. Caso de uso 2, validación de documentos.	29
Figura 3. Caso de uso 3, validación de inconsistencias.	30
Figura 4. Caso de uso 4, agendamiento.	30
Figura 5. Caso de uso 5, radicación en firme.	31
Figura 6. Caso de uso 6, validación de formatos.	32
Figura 7. Caso de uso 7, validación de inconsistencias.	32
Figura 8. Caso de uso 8, normalización de inconsistencias.	33
Figura 9. Caso de uso 9, trámite de bono.	33
Figura 10. Caso de uso 10, definición de la solicitud.	34

GLOSARIO

Apache: Servidor web de código abierto.

Su desarrollo comenzó en febrero de 1995, por Rob McCool, en una tentativa de mejorar el servidor existente en el NCSA. La primera versión apareció en enero de 1996, el Apache 1.0. Hacia el 2000, el servidor Web Apache era el más extendido en el mundo. El nombre «Apache» es un acrónimo de «a patchy server» -un servidor de remiendos-, es decir un servidor construido con código preexistente y piezas y parches de código. Es la auténtica «kill app» del software libre en el ámbito de los servidores y el ejemplo de software libre de mayor éxito, por delante incluso del kernel Linux. Desde hace años, más del 60% de los servidores web de Internet emplean Apache.

Aplicación Informática: Programa informático que permite a un usuario utilizar una computadora con un fin específico. Las aplicaciones son parte del software de una computadora, y suelen ejecutarse sobre el sistema operativo.

Asesoría: La asesoría es un servicio que consiste en brindar información a una persona real o jurídica. Médiante la misma se busca dar un respaldo en un tema que se conoce con gran detalle, respaldo que tiene una contrapartida en el hecho de garantizar un manejo eficiente de una situación determinada.

Back-up: Copia de seguridad de los ficheros o aplicaciones disponibles en un soporte magnético, con el fin de poder recuperar la información y las aplicaciones en caso de una avería en el disco duro, un borrado accidental o un accidente imprevisto. Es conveniente realizar copias de seguridad a intervalos temporales fijos, en función del trabajo y de la importancia de los datos manejados.

Base de datos: Conjunto de datos relacionados que se almacenan de forma que se pueda acceder a ellos de manera sencilla, con la posibilidad de relacionarlos, ordenarlos en base a diferentes criterios, etc. Las bases de datos son uno de los grupos de aplicaciones de productividad personal más extendidos. Entre las más conocidas pueden citarse dBase, Paradox, Access y Aprox, para entornos PC, y Oracle, ADABAS, DB/2, Informix, Ingres o Postgresql, para sistemas medios y grandes.

HTML (HyperText Markup Language): Lenguaje de marcado de Hipertexto. Es el lenguaje estándar para describir el contenido y la apariencia de las páginas en el WWW.

Linux: Sistema operativo de software libre (no es propiedad de ninguna persona o empresa), por ende no es necesario comprar una licencia para instalarlo y utilizarlo en un equipo informático. Es un sistema multitarea, multiusuario, compatible con

UNIX, y proporciona una interfaz de comandos y una interfaz gráfica, que lo convierte en un sistema muy atractivo y con estupendas perspectivas de futuro.

PHP (Preprocessed Hypertext Pages): PHP es un lenguaje de scripting embebido en HTML. Mucha de su sintaxis es tomada de C, Java y Perl con un par de características adicionales únicas y específicas de PHP. El propósito del lenguaje es permitir que los desarrolladores web escriban páginas generadas dinámicamente con rapidez.

Postgresql: Sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente.

Servidor: Computadora conectada a una red que pone sus recursos a disposición del resto de los integrantes de la red. Suele utilizarse para mantener datos centralizados o para gestionar recursos compartidos. Internet es en último término un conjunto de servidores que proporcionan servicios de transferencia de ficheros, correo electrónico o páginas WEB, entre otros.

Sistema de Pensiones: El Sistema General de Pensiones tiene por objeto garantizar a la población, el amparo contra las contingencias derivadas de la vejez, la invalidez y la muerte, mediante el reconocimiento de las pensiones y prestaciones que se determinan en la presente ley, así como propender por la ampliación progresiva de cobertura a los segmentos de población no cubiertos con un sistema de pensiones.

WorkFlow: Es la automatización de un proceso del negocio, entero o en partes, durante las cuales los documentos, la información o las tareas se pasan a partir de un participante¹ a otro para la acción, según un sistema de reglas procesales.

¹ * Recurso (ser humano o máquina)

RESUMEN

Este trabajo tiene por objeto mostrar cómo ha sido el desarrollo e implementación de un sistema de software a la medida para el control de las solicitudes de pensión, en este documento se evidencia cada uno de los pasos llevados a cabo para dicha implementación y se encuentra la documentación del sistema como tal, también se detallan los beneficios que ha traído al cliente la implementación de dicho proyecto, las metas de crecimiento y las expectativas futuro con el mismo.

Palabras clave

Flujo de trabajo, implementación, pensión, software, trámites, requerimientos, base de datos.

ABSTRAC

This work aims to show how it has been the development and implementation of a software system tailored to control pension applications, this document is evidence each of the steps taken for such implementation and is the system documentation as such, also details the benefits it has brought to the client the implemntación of the project, the goals of growth and future expectations with it.

Keywords

Workflow, implementation, pension, software, procedure, requerimientos, database.

1. INTRODUCCIÓN

A lo largo de este trabajo encontraremos la información detallada del proyecto de desarrollo de software, que se realizó a manera de pasantía para la empresa Imagine SAS y para su cliente Protección Pensiones y Cesantías, este proyecto tiene por efecto mejorar la situación de los usuarios que realizan tramites de pensión mediante la implementación de una solución de software a la medida, y es de vital importancia para el cliente ya que según estudios de mercado y análisis de su operación han encontrado que durante los próximos 10 años las solicitudes de pensión se incrementarán exponencialmente, situación que hace necesario realizar cambios a nivel operativo, de atención al cliente y, además de actualizaciones a nivel tecnológico y de infraestructura.

Este proyecto nace de la necesidad que tiene Protección de mejorar sus estándares de servicio y de controlar la operación relacionada con todo lo concerniente a las solicitudes de pensión de sus afiliados.

Este documento detalla las bondades en la ejecución del proyecto y sus beneficios en un corto, mediano y largo plazo.

2. JUSTIFICACIÓN

El proyecto avance de nuestro cliente Protección Pensiones y Cesantías se enfoca en la creación de un modelo de mejoramiento de servicio y actualización tecnológica, que permitirá reducir los tiempos de recepción, análisis y definición de las solicitudes de pensión, este proyecto se centra en prestar un servicio más humano teniendo en cuenta que la mayoría de las solicitudes de pensión tienen que ver con personas de edad avanzada, con familias que han perdido un ser querido o con personas en condición de invalidez, para mejorar este proceso se plantea establecer unos controles que permitan validar la calidad y completitud de los documentos solicitados para la solicitud, de esta manera el cliente podrá adjuntar los documentos mediante un correo electrónico y tendrá por esta misma vía respuesta de esta validación, esto evitará desplazamientos de estas personas y beneficiará a las oficinas de servicio con descongestión de sus actividades, de la misma forma el hecho de hacer un agendamiento personalizado permitirá tener un mayor control del servicio que se entrega al cliente en un tiempo menor y de mejor calidad, a nivel tecnológico el uso de un sistema de tipo workflow (flujo de trabajo) que permite validar las etapas del proceso y restringir ciertas actividades que impiden procesos posteriores.

Del mismo modo este flujo de trabajo permite integraciones entre sistemas que permiten automatizar procesos evitando una actuación humana y por ende minimizan el riesgo de error, también permite medir en línea el estado de una solicitud, lo que se traduce en mejor toma de decisiones ya que dependiendo de las etapas donde se presenten mayores represamientos, se pueden tomar acciones con miras a avanzar más rápido en el proceso, estas mediciones también permiten modificar las estrategias del negocio lo que deriva en una mejor condición para efectuar el trabajo y en un mejor servicio de cara al cliente.

Este proyecto es novedoso ya que implicó una preparación de más de 1 año, en el cual todas y cada una de las áreas que intervienen en la definición de una solicitud se juntaron para entregar sus puntos de vista, sus necesidades y las fortalezas de cada una al momento de operar, el juntar estas áreas permitió crear un modelo solido que considera las múltiples variables presentadas en este proceso y que decantó en un modelo tecnológico de desarrollo de software ágil que redujo los tiempos de desarrollo por el buen conocimiento y claridad de los objetivos del proyecto.

3. OBJETIVOS

3.1. Objetivo General

Desarrollar un modelo tecnológico y de procesos que permita reducir el tiempo de procesamiento de solicitudes de pensión según los estándares legales.

3.2. Objetivos Específicos

- Implementar un modelo de negocio que tenga en cuenta las necesidades de las diferentes áreas de la compañía para permitir mejorar los tiempos de respuesta a las solicitudes.
- Implementar mecanismos de medición que permitan controlar la operación y validen los resultados del nuevo modelo de negocio en línea.
- Desarrollar la estabilización de los nuevos procesos y efectuar nuevamente una convocatoria a las áreas para validar la aceptación del nuevo modelo de trabajo.
- Diseñar una base de datos relacional para el almacenamiento de las solicitudes de pensión y la trazabilidad de la operación realizada sobre las mismas.

4. MARCO TEÓRICO

Este marco teórico se centra en la funcionalidad de los sistema tipo workflow en la automatización y control de procesos, también resalta la importancia del control de las tareas realizadas dentro de un flujo de trabajo ya que permiten organizar los sistemas a partir de las reglas de negocio y facilitar a cada una de las áreas involucradas en un proceso ya que se resuelven las dependencias de sus labores, situación que facilita la ejecución de la labor diaria de los usuarios finales, así como también permite realizar mediciones en tiempo real y llevar trazabilidad de los procesos y de quien cuando y en qué orden se ejecutan.

Nota importante, las definiciones de este marco teórico son extraídas de diferentes fuentes y son citadas textualmente, el uso de esta información es netamente académico y de carácter informativo.

4.1. Definición

4.1.1. Primera Definición

El concepto de workflow en la historia moderna se remonta a 1912 cuando F. Taylor y a H. Gantt, juntos iniciaron el estudio de la organización racional de trabajo, referido al ámbito de la manufactura. Sin embargo, la acepción que nos interesa es la referida a sistemas informáticos de modo que incluyo la que utiliza la workflow.org, que dice:

“Workflow es la automatización de un proceso de negocio, sea parcial o totalmente, durante el cual documentos, información o tareas son pasados desde un participante a otro para la ejecución de otra acción, de acuerdo a un conjunto de reglas procedurales”.

Un Workflow debe permitir establecer un mecanismo de control que permita asegurar una secuencia de acciones y que estas acciones las ejecuten determinadas personas en un plazo pre-establecido.

4.1.2. Segunda definición

El workflow se define como un sistema informático que organiza y controla tareas, recursos y reglas, necesarias para completar el proceso de negocio, también se puede definir como la ruta que se debe seguir para la ejecución de una tarea o de un trabajo determinado, generalmente trabaja en secuencia y funciona como una línea de producción.

4.2. Ventajas de un workflow

- Permite organizar las tareas.
- Permite realizar planificaciones desde el inicio del proceso o tarea a ejecutar.
- Permite realizar mediciones en cada una de las etapas.
- Soluciona las dependencias de una tarea, por ser secuencial (una tarea no se inicia hasta no acabar sus etapas previas).
- Mejoras notables en la productividad.
- Mejora la relación con los clientes ya que por su flexibilidad permite saber en el momento exacto, la etapa exacta de una tarea.
- Mejora el control de procesos, permite encontrar cuellos de botella en las tareas ejecutadas día a día.
- Permite automatizar tareas repetitivas.

4.3. Integración

Una característica fundamental de un sistema Workflow es la correcta integración con los sistemas de información actuales, como Bases de Datos, Gestión Documental, Mensajería, ERP, GroupWare, Call Centers, Mainframe, etc ... Existen firmas especializadas de sistemas que llevan años integrando soluciones basadas en esta tecnología, garantizando la correcta ejecución de cualquier proyecto.

4.4. Introducción a programas Workflow

Hoy en día, es vital acceder a la información de un modo ágil y eficaz. Normalmente esta información se encuentra en diferentes formatos (papel, imágenes, archivos informáticos etc.). Al estar en diferentes formatos su accesibilidad no es rápida y fiable lo que sería deseable.

Actualmente es impresionante la pérdida de tiempo en la localización de información en la empresa. Gracias a los sistemas de gestión documental, este tiempo (días útiles al año), se convierte en pocos segundos. Este tiempo ahorrado se traduce en una importante reducción de costes e incremento en la productividad de los usuarios o empleados.

Mientras que la automatización ha conseguido aumentar la eficacia en los procesos industriales en órdenes de magnitud. En los procesos administrativos, este aumento se ha cifrado en sólo 3%, frente a la realización manual. Existen en la actualidad herramientas nuevas que van a permitir conseguir incrementos. Clasificadas dentro del paquete genérico de "herramientas de trabajo en grupo", incluyen piezas claves

de automatización de procesos administrativos (Workflow) y de gestión de documentación.

Los sistemas de Workflow permiten soportar los clásicos circuitos de expedientes directamente sobre una red de ordenadores, recurriendo al papel sólo cuando es necesario. Por medio de un "motor de Workflow" situado en el servidor de la red, van encaminando los expedientes de acuerdo con las reglas establecidas, y facilitan las tareas de los usuarios integrando en el circuito formularios electrónicos, acceso a bases de datos y ordenadores centrales, así como el enlace con otras aplicaciones ofimáticas. Además, los buenos paquetes de Workflow permiten la programación de las reglas con un mínimo de programación, habitualmente con herramientas gráficas.

Las herramientas de gestión documental permiten mantener toda la información sin soporte de papel, sobre una red o sistema informático. Los documentos son escaneados y almacenados en discos de gran capacidad, y los sistemas ofrecen potentes utilidades para analizar, gestionar y localizar toda la información almacenada.

Estas herramientas se pueden integrar entre sí, así como sobre los nuevos sistemas de trabajo en grupo, tales como Lotus Notes o MS Exchange, que añaden las facilidades de coordinación y comunicación entre equipos de trabajo, soportado sobre bases de datos, y facilitando, si es necesario, el acceso vía inter o intranet.

4.5. Workflow en el sector financiero

Los suministradores de aplicaciones informáticas para el sector financiero también reconocen la importante ventaja competitiva que obtienen incorporando herramientas de WORKFLOW a sus productos. Guadra Computer Services Ltd, entre otras, ofrece un amplio abanico de aplicaciones para la banca hipotecaria y personal y ha mejorado toda su gama (conocida como SUMMIT) incorporando funcionalidades de Workflow STAFFWARE. Estas nuevas capacidades controlan y definen los procesos del sistema de administración de clientes. La reingeniería de este producto ha permitido un alto grado de flexibilidad separando lo que es la aplicación de lo que es el control de las secuencias y el control de las actividades unitarias del proceso general. De este modo, los usuarios de SUMMIT, disponen de nuevas e interesantes posibilidades en su aplicación:

- Hanley Economic Building Society.
- Cumberland Building Society.
- Stroud & Swindon Building Society.
- Tipton & Coseley Building Society.

4.5.1. Olivetti

Olivetti ha integrado la tecnología Workflow de STAFFWARE dentro de sus productos Pinnacle Plus y Mosaic OA. Estos dos productos son la base de todas las aplicaciones de Olivetti que proporcionan soluciones para la gestión automática del front y back office.

4.5.2. Staffware

Mercantile Mutual: Se trata de un destacado grupo de empresas de servicios financieros y de seguros y, gracias a su nuevo sistema de gestión documental y la implantación de un Workflow de STAFFWARE, está en condiciones de proporcionar repuesta inmediata a las peticiones de sus clientes. La instalación comenzó en la Retail Funds Management Division en noviembre de 1995.

El nuevo sistema STAFFWARE permite a los empleados del servicio de atención al cliente, acceder a cualquier documento previamente almacenado en cuestión de segundos.

El tratamiento de imágenes documentales junto con Workflow de STAFFWARE aporta un nuevo punto de vista para la reingeniería de los procesos internos. Muy pocas veces se evalúa el coste de las búsquedas clásicas en archivos de papel y de la devolución de llamadas previas de clientes pero es aún más difícil poder cuantificar los efectos que los retrasos en la gestión tienen en la relación con los clientes. El sistema para 50 usuarios instalado en Mercantile Mutual tiene por objetivo proporcionar una verdadera ventaja competitiva a través de la diferenciación por la calidad de servicio.

"La calidad del servicio al cliente es la clave de éxito de una empresa. Un control eficaz de este servicio, personalizándolo y haciéndolo acorde con las necesidades de los clientes será un poderoso diferenciador en lo que queda de esta década. "

4.5.3. Bruce Appleton, Imaging & Workflow Manage Mercantile Mutual

Uno de los mayores sistemas, por volumen manejado, de workflow STAFFWARE y tratamiento de imágenes, está totalmente operacional en LASB Local Authorities Superannuation Board. El sistema fue instalado en un tiempo record de 10 semanas y recoge, controla y realiza inversiones manejando un montante de hasta 1,3 billones de dólares, perteneciente a los 73.000 miembros del Board. La aplicación utiliza "scanners" de alta velocidad y códigos de barras durante la fase de gestión de los documentos para capturar los datos correspondientes a los socios. El producto STAFFWARE controla el tratamiento de los ficheros así creados. Pero el

proyecto de Superannuation representa solo el comienzo dentro de LASB que planea implementar STAFFWARE en toda su organización.

4.6. Otras aplicaciones de los Workflows

4.6.1. Workflows Consuetudinarios

Son las secuencias de operación que se establecen simplemente por el uso y la costumbre.

4.6.2. Woorkflows Manuales

Son los que se establecen en un documento que contiene reglas, secuencias de operaciones, listas de datos posibles, identificación de los V°B°, etc. y también incluyen un diagrama de flujo del proceso. Este tipo de documento es lo que se llama Procedimiento. En este procedimiento pueden intervenir sistemas computacionales como ser: correo electrónico, planillas de cálculo, ERP, etc. El primer caso que recuerdo de este tipo de workflows fue el de Órdenes de Compra de Importación que se implementó usando correo electrónico en Sonda en la década de los 80.

4.6.3. Workflows Implícitos

Estas son secuencias de operación que la lógica de un sistema sugiere utilizar, normalmente corresponden a un proceso de negocios, por ejemplo la secuencia: Cotización -> Nota de Venta -> Picking -> Despacho -> Factura -> Pago. Los llamo implícitos porque permiten establecer una secuencia entre una transacción y la siguiente, pero no controlan que persona es responsable de ejecutar cada transacción y en que período de tiempo. Casos como este los ví en algunos sistemas ERP como Microsoft Solomon (hoy es Dynamics) y SAP Business One.

4.6.4. Workflow Documentales

Estas son secuencias típicamente relacionados con un documento, sea para la aprobación de su distribución o para autorizar algún tipo de operación. Por ejemplo: autorizar la publicación de una nueva norma, definir la creación de un nuevo código de productos. Estos workflow se generan con el soporte de algún software como ser Lotus, SAP KM, etc. que básicamente controlan la secuencia de ejecución y operan sobre un formulario, el que van pasando de persona en persona, sea para completar datos o para dar autorizaciones (V°B°).

4.6.5. Workflow de Aplicaciones

Son aquellos que vienen incluidos en una aplicación con secuencias pre-establecidas y con la posibilidad de parametrizarlos, esto los he encontrado en los ERP y son los que claramente corresponde a un workflow según lo plantean las organizaciones que están definiendo los estándares para los workflows, Estos se encuentra en sistemas como SAP-ECC (ERP Central Component).

4.6.6. Workflow ad-hoc

Estos son los que una organización puede utilizar para diseñar y construir a la medida, se supone que la ventaja competitiva las corporaciones hoy la obtienen por medio de la innovación en sus procesos de negocios, es decir en la manera como ejecuta sus operaciones. y es aquí donde entran con fuerza los Workflows porque con ellos es posible establecer secuencias de operación controlables y medibles. Actualmente estoy con un grupo de colegas explorando SAP Guided Procedures para desarrollar estos Workflows ad-hoc.

Para este proyecto se desarrollo un workflow manual con la siguiente secuencia de procesos:

- Captura de Solicitudes
- Recepción de Documentos
- Validación de Documentos
- Validación de Inconsistencias en Información del afiliado
- Agendamiento del Caso
- Radicación del Caso
- Validación de formatos
- Control de Calidad a Validación de Inconsistencias
- Normalización de Inconsistencias
- Tramite de Bono Pensional
- Analisis De La Solicitud
- Definición de la prestación.

5. INGENIERÍA DEL PROYECTO

5.1. Descripción de la Situación Actual

Protección Pensiones y Cesantías cuenta con varios sistemas de información que no convergen entre ellos y que conlleva a una operación manual bastante grande, la compañía cuenta con una herramienta core de la compañía que se llama AS400 y un sistema de Microsoft SharePoint que alimenta al AS400, adicional a esto cuenta con software de terceros para llevar un control de las imágenes y documentos de las solicitudes de pensión, el sistema SharePoint tiene definidos flujos de negocio diferentes a los definidos en la estructuración inicial del AS400, cambiar estas reglas de negocio implicaría una inversión bastante fuerte para la compañía en dinero y en tiempo de desarrollo, como las reglas de negocio no concuerdan hay una problemática con algunos registros que quedan en un “limbo” ya que en el sistema SharePoint aparecen como cargados al AS400, pero en el AS400 no están los registros, se requiere de una comparación diaria de registros para establecer cuales registros y por qué no llegaron al sistema principal, además de esto se requiere que los responsables del SharePoint hagan actualizaciones y cargues manuales directamente a las tablas del sistema principal.

También es importante anotar que la compañía cuenta con un panel de analistas legales para la validación de todas y cada una de las solicitudes de pensión, pero el control a nivel de sistema para la oportunidad y la calidad de este análisis se realiza en Excel.

Otro caso actual es que como hay varios sistemas no se puede hacer un tracking online de las solicitudes, por este motivo los canales de atención a clientes no pueden informar como es debido a los afiliados de en qué momento y sobre todo cuanto tiempo más tardará su solicitud de pensión en procesarse.

Adicional a esto se están incumpliendo los tiempos de ley para el análisis de estas solicitudes, ya que no es posible con el método actual de trabajo establecer los cuellos de botella de las solicitudes ni las áreas que tienen más problemas para solucionar las inconsistencias en una solicitud.

5.2. Requerimientos de la Información

5.2.1. Requerimientos funcionales

5.2.1.1. Asesoría Preliminar

El usuario debe contar con un formulario en una aplicación web que le permita capturar la información de las solicitudes de pensión a través de los diferentes canales de atención a clientes (call center, oficinas de servicio) cada trámite se debe identificar con un tipo de solicitud previamente entregado y debe tener parametrizado de acuerdo con el tipo de solicitud un cuestionario que permita filtrar los documentos requeridos para el trámite de pensión, al final de la captura de datos, el sistema debe generar un documento pdf que entregue un reporte de la información capturada y el listado de documentos que apliquen para la solicitud.

5.2.1.2. Carga Masiva de Documentos

Cuando el usuario capture la información de la solicitud, el sistema debe activar una tarea de recepción de los documentos solicitados al afiliado, y una tarea para establecer la categoría del trámite como (PREAPROBADO, MASIVO O COMPLEJO), el sistema también, debe habilitar una carga masiva de los documentos recibidos por trámite y debe activar la validación de los mismos.

5.2.1.3. Validación Documental

Para la validación de documentos el sistema debe proveer un listado de los documento solicitados en el trámite y debe mostrar en un visor los documentos recibidos del afiliado, el sistema muestra una lista de chequeo y el usuario marca los documentos recibidos o no, en caso de que hagan falta documentos o estos no cumplan con la calidad solicitada se debe activar una tarea para realizar una llamada al afiliado y mostrar los documentos que tiene pendientes en su trámite, por el contrario si los documentos están completos, se activa una tarea para realizar una llamada al afiliado y para que se programe una agenda para realizar la radicación en firme del trámite.

5.2.1.4. Radicación de Solicitudes

Se debe visualizar una pantalla que permita realizar la radicación en firme del trámite de pensión, en esta pantalla se ingresará el número de cédula y el tipo de solicitud a radicar, el sistema debe consultar en la base de datos y cargar el número de solicitud, cuando el usuario radique el caso, el sistema debe generar una tarea de validación de formatos similar a la validación de documentos y debe volver a activar la tarea de validación de inconsistencias nuevamente, el trámite debe quedar con estado radicado.

5.2.1.5. Categorización de Solicitudes

El sistema tendrá parametrizadas una serie de combinaciones que permitirán clasificar los trámites dentro de una categoría (PREAPROBADO, MASIVO Y COMPLEJO), el sistema debe crear las tareas respectivas en las pantallas que presente inconsistencias y si no existen las mismas debe activar la etapa de definición del trámite.

5.2.1.6. Solución de Inconsistencias

Los usuarios de cada área deben tener una bandeja que les permita listar los tramites que tengan inconsistencias relacionadas con cada una de sus áreas, en el sistema los usuarios indicaran si la inconsistencias fue solucionada, el sistema llevara el registro de fechas de solución de cada inconsistencia y cuando se encuentren solucionadas todas las inconsistencias detectadas activará la tarea de definición del trámite de pensión.

5.2.1.7. Definición del Trámite

El usuario debe tener una pantalla que permita registrar la definición del caso, se debe poder guardar las gestiones realizadas, tales como los cobros y pagos al estado por concepto del trámite de pensión, también debe existir una pantalla en la que el usuario registré que se envió la notificación al afiliado de su trámite de pensión ya sea por correo certificado, llamada o cita en la oficina de servicio.

5.2.1.8. Reportes de la aplicación

El sistema debe generar un reporte general indicando que tareas se activaron para cada etapa del flujo, las fechas de inicio y cierre de tareas, así como los usuarios involucrados en cada etapa, el reporte debe generarse en Excel y debe permitir filtrar por etapa del flujo, por fecha de creación de la solicitud, por cédula y por número de trámite.

5.2.1.9. Ficha General

El usuario dispondrá de una consulta en línea que permitirá ver el histórico de gestión de un caso específico, la etapa actual en la que se encuentra y la fecha proyectada de definición del caso, esta ficha debe permitir consultar por cédula del afiliado y por número de trámite.

5.2.2. Requerimientos no funcionales

5.2.2.1. Requerimientos de Sistema Operativo

La aplicación corre bajo un esquema cliente servidor con un sistema operativo distribución Linux Suse Enterprise server 10 o superior.

5.2.2.2. Perfilamiento

La aplicación debe controlar que los usuarios accedan solo a la información que les corresponde visualizar.

5.2.2.3. Lenguaje de Programación

El sistema debe permitir consultarse en la intranet interna de la compañía, para dicho fin se implementará en lenguaje de programación Php versión 5 o superior y servidor web apache 2 prefork.

5.2.2.4. Almacenamiento de Datos

El sistema debe almacenar los datos en una base de datos relacional y tener un esquema de backup diario.

5.2.2.5. Soporte a la aplicación

El soporte de la aplicación debe ser 7 x 24, esto incluye soporte de backups y alta disponibilidad con servidor de contingencia.

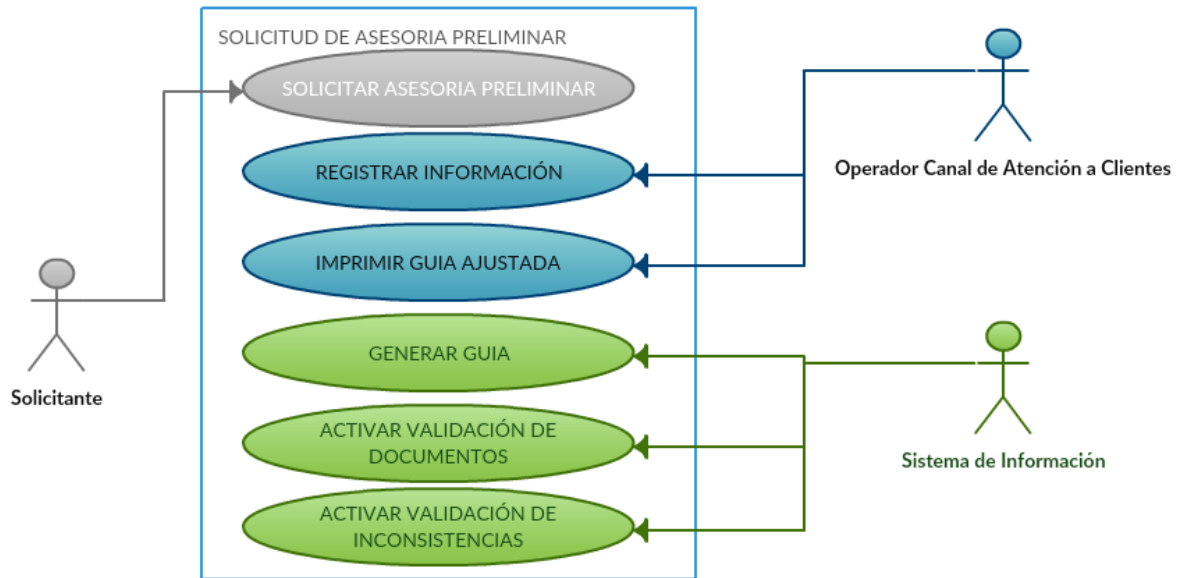
5.3. Modelamiento del Sistema

5.3.1. Diagramas de Casos de Uso

5.3.1.1. Caso 1 – Solicitud Asesoría preliminar

En este caso el afiliado establece comunicación con alguno de los canales de servicio de Protección (Call center, Oficina de Servicio) y realiza la solicitud de asesoría preliminar, el asesor de servicio registra la información en el sistema y este entrega una guía con los documentos requeridos para iniciar el trámite de pensión, el sistema queda a la espera de que el afiliado entregue estos documentos.

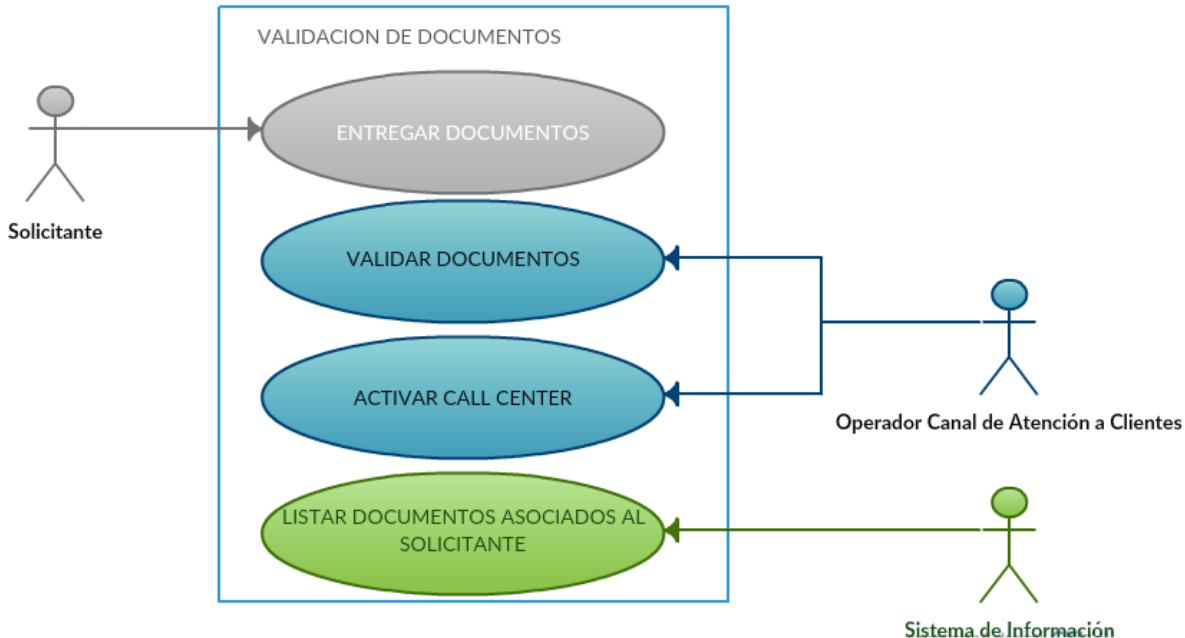
Figura 1 - Caso de uso 1, solicitud de asesoría preliminar.



5.3.1.2. Caso 2 – Validación de Documentos

En este caso una vez recibidos los documentos por parte del afiliado el sistema genera una lista de chequeo para validar la completitud de los documentos requeridos, si hay documentación pendiente se activa al call center para que efectúe una llamada solicitando el documento o su corrección, si los documentos se encuentran completos el sistema activa al call center para que efectúe el agendamiento de la cita para radicar en firme la solicitud de trámite de pensión.

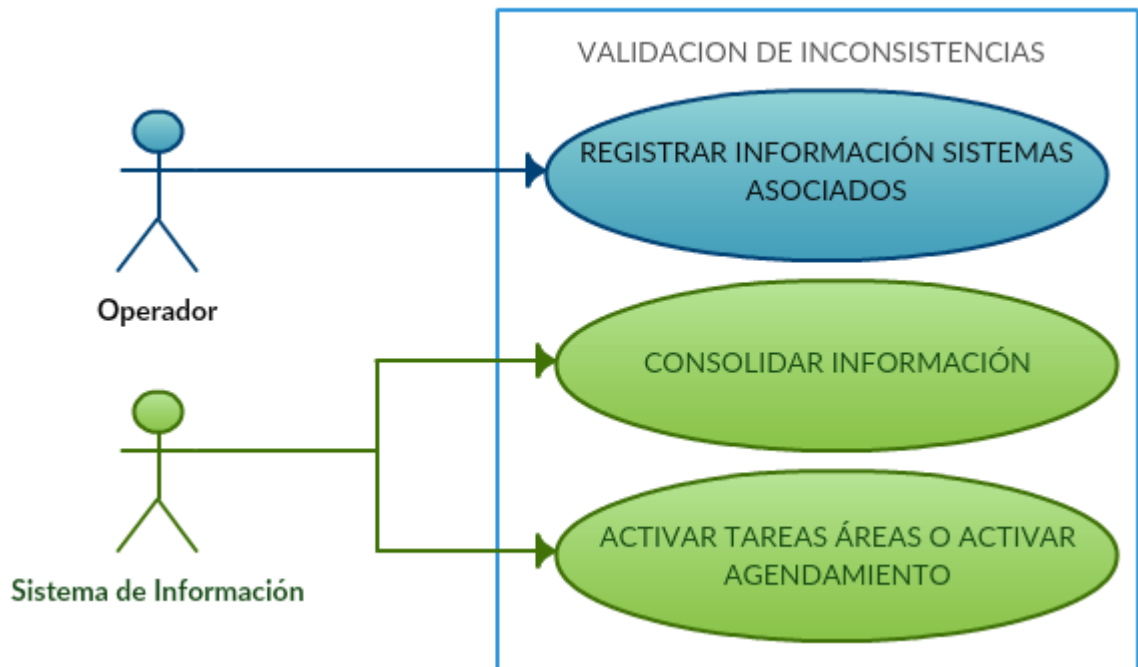
Figura 2 - Caso de uso 2, validación de documentos.



5.3.1.3. Caso 3 – Validación de Inconsistencias

En ese caso el sistema habilita un formulario para que el operador realice una captura de los datos personales del afiliado en diferentes sistemas externos (Siafp, OBP y AS400) una vez capturados los datos, el sistema compara los datos versus lo almacenado en la solicitud de pensión, si se presentan diferencias activa a las áreas encargadas de actualizar dicha información para que hagan la corrección correspondiente, el sistema también entrega una lista de chequeo en la que se registra si el afiliado presenta inconsistencias en su historia laboral, pagos o deudas con la compañía, el sistema activa a cada una de las áreas encargadas de gestionar dicha información, si el caso no presenta inconsistencias el sistema activa la definición de la prestación.

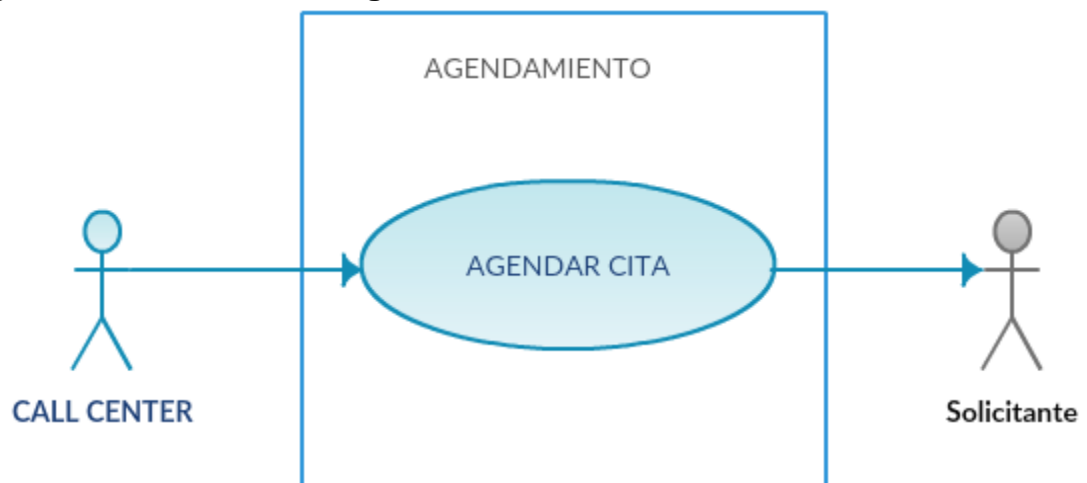
Figura 3 - Caso de uso 3, validación de inconsistencias.



5.3.1.4. Caso 4 – Agendamiento

En el momento en que el afiliado tiene la documentación requerida para el trámite de pensión completa, se habilita una agenda en el sistema para que radique en firme su trámite.

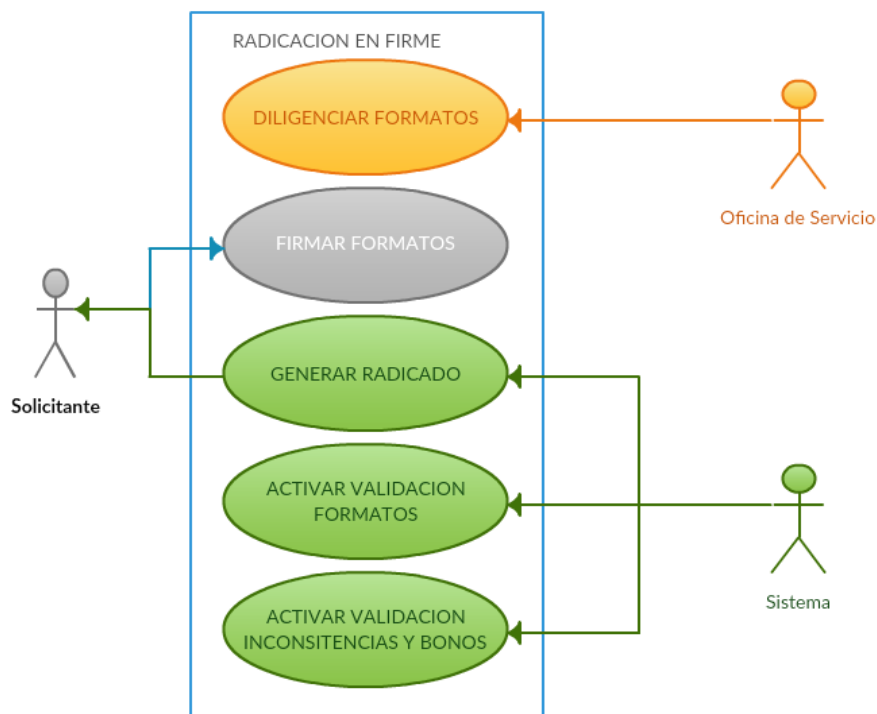
Figura 4 - Caso de uso 4, agendamiento.



5.3.1.5. Caso 5 – Radicación en firme

En este caso el afiliado cumple su agenda y se procede a la radicación en firme de la solicitud, este paso es importante porque a partir de este punto se empiezan a contar los tiempos de ley en los que un fondo de pensión debe dar respuesta a una solicitud de pensión, en este caso el operador del sistema busca la solicitud por número de cédula y tipo de solicitud y activa una nueva validación de formatos y nuevamente la validación de inconsistencias ya que durante el periodo que transcurre entre la asesoría preliminar y la radicación pueden llegar a presentarse.

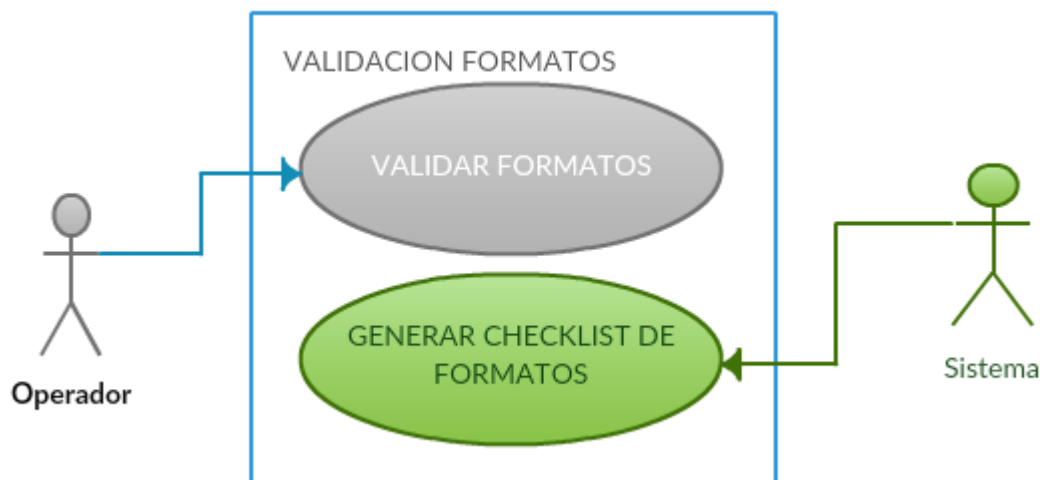
Figura 5 - Caso de uso 5, radicación en firme.



5.3.1.6. Caso 6 – Validación de Formatos

En este caso el sistema presenta nuevamente una lista de chequeo para que el operador valide si la oficina de servicio envió todos los formatos que debe diligenciar al momento de la radicación del trámite de pensión.

Figura 6 - Caso de uso 6, validación de formatos.



5.3.1.7. Caso 7 – Validación de Inconsistencias

Este caso opera igual que el literal 5.3.1.3., se debe tener en cuenta que si en la validación inicial se presentó una inconsistencia pero en esta validación no se presenta, el sistema debe cerrar la tarea de validación automáticamente.

Figura 7 - Caso de uso 7, validación de inconsistencias.

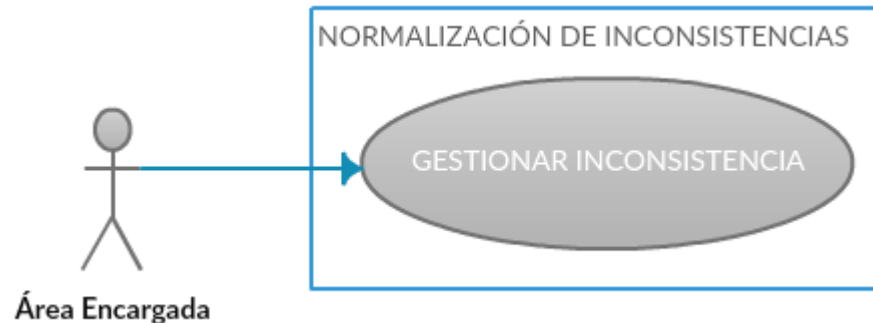


5.3.1.8. Caso 8 – Normalización de Inconsistencias

En este caso cada una de las áreas activadas con alguna inconsistencia ingresan al sistema para notificar que se han solucionado dichas inconsistencias, el sistema calcula que todas las inconsistencias estén solucionadas y avanza al caso a la

segunda etapa de la validación de inconsistencias, que se llama validación de bonos.

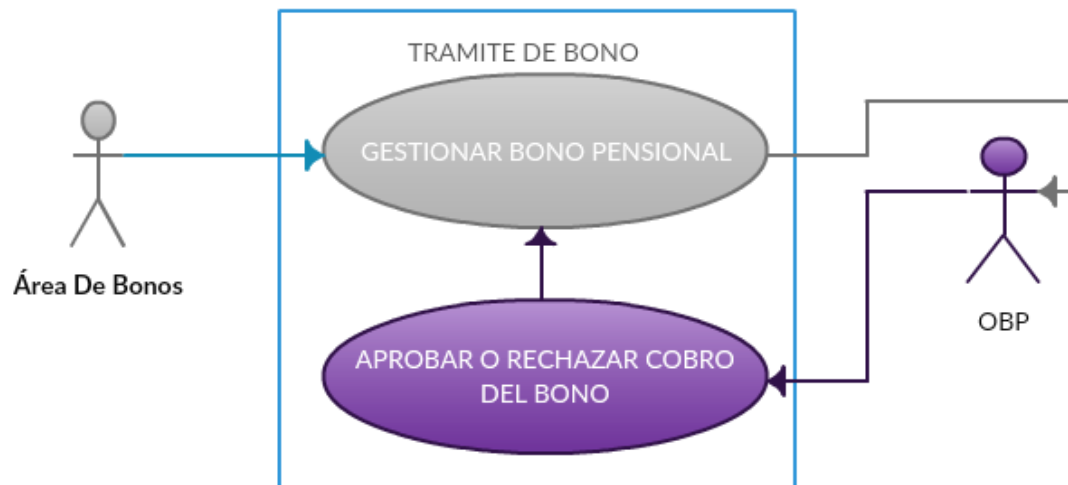
Figura 8 - Caso de uso 8, normalización de inconsistencias.



5.3.1.9. Caso 9 - Trámite de Bono

Este caso es la segunda etapa en la normalización del trámite, en este caso se validan y se realizan los cobros requeridos entre fondos de pensiones y gubernamentales para garantizar que las semanas de pensión se encuentren completas para la definición del trámite de pensión.

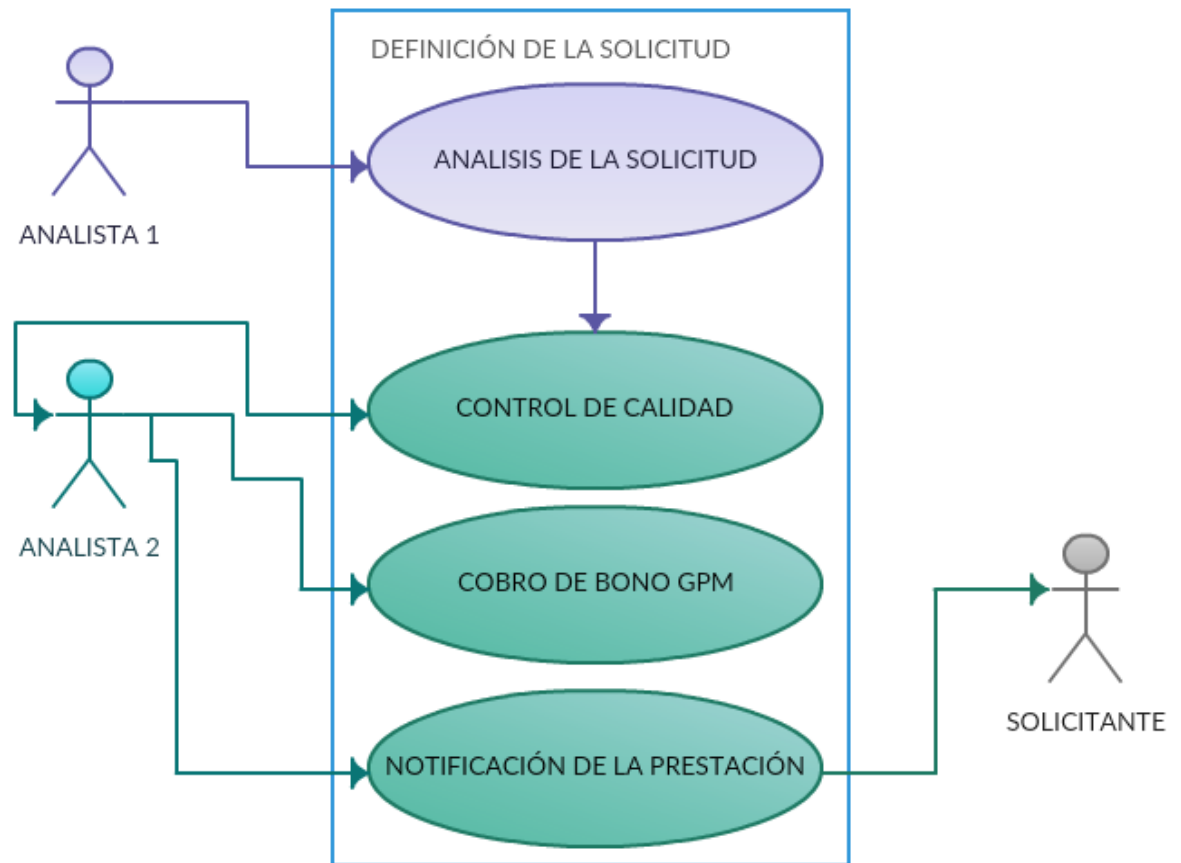
Figura 9 - Caso de uso 9, trámite de bono.



5.3.1.10. Caso 10 – Definición de la solicitud

Esta es la etapa final del flujo, en esta etapa un panel de expertos realiza una serie de validaciones para definir si hay derecho a pensión, cuál será el monto y entre que beneficiarios se entrega la pensión, también hay una módulo de notificación al solicitante de pensión.

Figura 10 - Caso de uso 10, definición de la solicitud.



5.3.2. Script de la base de datos.

```
CREATE DATABASE prestaciones_test
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1;
CREATE TABLE preguntas(
cod_pregunta character varying NOT NULL,
pregunta character varying,
estado integer,
codigo_agrupacion character varying,
orden integer,
CONSTRAINT pk_preguntas_cod_pregunta PRIMARY KEY (cod_pregunta)
)WITH ( OIDS=FALSE );
ALTER TABLE preguntas OWNER TO postgres;

CREATE TABLE imagenes_asociadas(
id serial NOT NULL,
id_imagen integer,
tramite_nuevo character varying,
fecha_asociacion timestamp without time zone DEFAULT now(),
usuario_asocia character varying,
CONSTRAINT pk_imagenes_asociadas_id PRIMARY KEY (id)
) WITH ( OIDS=FALSE );
ALTER TABLE imagenes_asociadas OWNER TO postgres;

CREATE TABLE acciones_legales(
id serial NOT NULL,
nombre_accion character varying,
cierra_tareas character varying,
CONSTRAINT pk_acciones_legales_id PRIMARY KEY (id)
) WITH ( OIDS=FALSE );
ALTER TABLE acciones_legales OWNER TO postgres;

CREATE TABLE acuerdos_servicio(
na serial NOT NULL,
tipo_solicitud character varying,
mesa_control1 integer,
precategorizacion integer,
mesa_control2 integer,
categorizacion integer,
validaciones_sitio integer,
liquidaciones integer,
historias_laborales integer,
llamada_funeraria integer,
crear_ben_grupos integer,
precategorizacion_grupos integer,
categorizacion_grupos integer,
simulacion integer,
CONSTRAINT pk_acuerdos_servicio_na PRIMARY KEY (na)
) WITH ( OIDS=FALSE );
ALTER TABLE acuerdos_servicio OWNER TO postgres;
```

```

CREATE TABLE adj_conceptos(
    id_adjunto serial NOT NULL,
    id_concepto integer,
    path_adjunto character varying,
    CONSTRAINT adj_conceptos_pkey PRIMARY KEY (id_adjunto)
)WITH ( OIDS=FALSE);
ALTER TABLE adj_conceptos OWNER TO postgres;
ALTER TABLE adj_conceptos ALTER COLUMN id_adjunto SET STATISTICS 0;
ALTER TABLE adj_conceptos ALTER COLUMN id_concepto SET STATISTICS 0;
ALTER TABLE adj_conceptos ALTER COLUMN path_adjunto SET STATISTICS 0;

CREATE TABLE adj_dictamenes(
    id serial NOT NULL,
    id_concepto integer,
    path character varying,
    CONSTRAINT pk_adj_dictamen PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE adj_dictamenes OWNER TO postgres;

CREATE TABLE agrupaciones(
    id character varying NOT NULL,
    nombre_agrupacion character varying,
    CONSTRAINT pk_agrupaciones_id PRIMARY KEY (id)
) WITH ( OIDS=FALSE );
ALTER TABLE agrupaciones OWNER TO postgres;

CREATE TABLE asignacion_bonos (
    id serial NOT NULL,
    usuario character varying,
    tramite character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_asignacion_bonos_id PRIMARY KEY (id)
) WITH ( OIDS=FALSE );
ALTER TABLE asignacion_bonos OWNER TO postgres;

CREATE TABLE asignacion_gestion
(
    id serial NOT NULL,
    usuario character varying,
    tramite character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_asignacion_gestion_id PRIMARY KEY (id)
) WITH ( OIDS=FALSE );
ALTER TABLE asignacion_gestion OWNER TO postgres;

CREATE TABLE asignacion_imagine(
    id serial NOT NULL,
    tramite character varying,
    usuario character varying,
    bandeja character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_asignacion_imagine_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);

```

```

ALTER TABLE asignacion_imagine OWNER TO postgres;

CREATE TABLE asignacion_tramites(
    na serial NOT NULL,
    tramite character varying,
    usuario character varying,
    proceso character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    tipo_solicitud character varying,
    categoria character varying,
    CONSTRAINT pk_asignacion_tramites_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE asignacion_tramites OWNER TO postgres;

CREATE TABLE beneficiarios_prerradicacion(
    id_ben serial NOT NULL,
    cod_prerradicacion character varying,
    tipo_benef integer,
    tipo_sol character varying,
    identificacion_benef character varying,
    primer_nombre character varying,
    segundo_nombre character varying,
    primer_apellido character varying,
    segundo_apellido character varying,
    razon_social character varying,
    telefono character varying,
    celular character varying,
    autosms character varying,
    correo character varying,
    autoenvio character varying,
    ciudad character varying,
    tipodoc_beneficiario character varying,
    direccion_correspondencia character varying,
    fechanac_benef character varying,
    genero_ben character varying,
    calidad_benef character varying,
    parentesco character varying,
    user_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    estado_cruce character varying,
    CONSTRAINT pk_beneficiarios_prerradicacion PRIMARY KEY (id_ben)
)WITH ( OIDS=FALSE);
ALTER TABLE beneficiarios_prerradicacion OWNER TO postgres;

CREATE TABLE beneficiarios_prerradicacion_temp(
    id_ben serial NOT NULL,
    cod_prerradicacion character varying,
    tipo_benef integer,
    tipo_sol character varying,
    identificacion_benef character varying,
    primer_nombre character varying,
    segundo_nombre character varying,
    primer_apellido character varying,
    segundo_apellido character varying,

```

```

razon_social character varying,
telefono character varying,
celular character varying,
autosms character varying,
correo character varying,
autoenvio character varying,
ciudad character varying,
tipodoc_beneficiario character varying,
direccion_correspondencia character varying,
fechanac_benef character varying,
genero_ben character varying,
calidad_benef character varying,
parentesco character varying,
user_insert character varying,
fecha_insert timestamp without time zone DEFAULT now(),
CONSTRAINT pk_beneficiarios_prerradicacion_temp PRIMARY KEY (id_ben)
)WITH ( OIDS=FALSE);
ALTER TABLE beneficiarios_prerradicacion_temp OWNER TO postgres;

CREATE TABLE beneficiarios_temp(
id_temp character varying,
tipo_benef integer,
tiposol character varying,
subtipo character varying,
identificacion_benef character varying,
primer_nombre character varying,
segundo_nombre character varying,
primer_apellido character varying,
segundo_apellido character varying,
razon_social character varying,
telefono character varying,
celular character varying,
autosms character varying,
correo character varying,
autoenvio character varying,
ciudad character varying,
tipodoc_beneficiario character varying,
direccion_correspondencia character varying,
consecutivo serial NOT NULL
)WITH ( OIDS=FALSE);
ALTER TABLE beneficiarios_temp OWNER TO postgres;

CREATE TABLE casos_devueltos(
id serial NOT NULL,
tramite character varying,
bandeja character varying,
CONSTRAINT pk_casos_devueltos_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE casos_devueltos OWNER TO postgres;

CREATE TABLE causales_rechazo(
id_causal serial NOT NULL,
descripcion character varying,
estado_cau integer DEFAULT 0,

```

```

    tipo_solicitud character varying,
    CONSTRAINT pk_causales_rechazo_id_causal PRIMARY KEY (id_causal)
)WITH ( OIDS=FALSE);
ALTER TABLE causales_rechazo OWNER TO postgres;

CREATE TABLE cerradas_por_cierre(
    id serial NOT NULL,
    id_traza integer,
    motivo_cierre character varying,
    observacion text,
    fecha_cierre timestamp without time zone DEFAULT now(),
    estado integer DEFAULT 0,
    CONSTRAINT pk_cerradas_por_cierre_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE cerradas_por_cierre OWNER TO postgres;

CREATE TABLE ciudades(
    codigo character varying NOT NULL,
    codigo_depto character varying,
    nombre_depto character varying,
    codigo_ciudad character varying,
    nombre_ciudad character varying,
    CONSTRAINT pk_ciudades_codigo PRIMARY KEY (codigo)
)WITH ( OIDS=FALSE);
ALTER TABLE ciudades OWNER TO postgres;

CREATE TABLE clasificacion_tiposol(
    id_clas serial NOT NULL,
    codigo character varying,
    tipo_solicitud character varying,
    subtipo integer,
    letra_prerradicacion character varying,
    titulo_pdf character varying,
    encabezado text,
    notas text,
    CONSTRAINT pk_clasificacion_tiposol_codigo PRIMARY KEY (id_clas)
)WITH ( OIDS=FALSE);
ALTER TABLE clasificacion_tiposol OWNER TO postgres;

CREATE TABLE compara_vigencias(
    na serial NOT NULL,
    tramite character varying,
    fecha_efectividad_as character varying,
    fecha_efectividad_siafp character varying,
    fecha_efectividad_obp character varying,
    tipo_vinculacion_as character varying,
    tipo_vinculacion_siafp character varying,
    tipo_vinculacion_obp character varying,
    usuario_gestion character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_compara_vigencias_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE compara_vigencias OWNER TO postgres;

```



```

CREATE TABLE compara_vigencias_tmp(
    na serial NOT NULL,
    tramite character varying,
    fecha_efectividad_as character varying,
    fecha_efectividad_siafp character varying,
    fecha_efectividad_obp character varying,
    tipo_vinculacion_as character varying,
    tipo_vinculacion_siafp character varying,
    tipo_vinculacion_obp character varying,
    usuario_gestion character varying,
    CONSTRAINT pk_compara_vigencias_tmp_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE compara_vigencias_tmp OWNER TO postgres;

CREATE TABLE comparacion_efectividad_prerrequisitos(
    na serial NOT NULL,
    tramite character varying,
    plataforma character varying,
    fecha_efectividad character varying,
    tipo_vinculacion character varying,
    fecha_comparacion timestamp without time zone DEFAULT now(),
    usuario_comparacion character varying,
    CONSTRAINT pk_comparacion_efectividad_prerrequisitos_na PRIMARY KEY
(na)
)WITH ( OIDS=FALSE);
ALTER TABLE comparacion_efectividad_prerrequisitos OWNER TO postgres;

CREATE TABLE concepto_ips(
    id serial NOT NULL,
    pronositco character varying,
    usuario character varying,
    tramite character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    observaciones text,
    cruce_ips character varying
DEFAULT 'PENDIENTE CRUZAR'::character varying,
    fecha_cruce timestamp without time zone,
    dia_remision character varying,
    CONSTRAINT pk_concepto_ips_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE concepto_ips OWNER TO postgres;

CREATE TABLE contribuyentes(
    id serial NOT NULL,
    nit character varying,
    nombre character varying,
    clasificacion character varying,
    CONSTRAINT pk_contribuyentes_id PRIMARY KEY (id)
)WITH (OIDS=FALSE);
ALTER TABLE contribuyentes OWNER TO postgres;

CREATE TABLE cuestionario(
    id serial NOT NULL,
    clas character varying,

```

```

    pregunta character varying,
    CONSTRAINT pk_cuestionario PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE cuestionario OWNER TO postgres;

CREATE TABLE det_error_gest_prerrequisitos(
    na serial NOT NULL,
    tarea character varying,
    campo_error character varying,
    tramite character varying,
    usuario_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_det_error_gest_prerrequisitos_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE det_error_gest_prerrequisitos OWNER TO postgres;

CREATE TABLE documento_devolucion(
    tramite character varying NOT NULL,
    id_documento_faltante integer NOT NULL,
    id_traza integer NOT NULL,
    gestionado integer NOT NULL DEFAULT 0,
    observacion text
)WITH ( OIDS=FALSE);
ALTER TABLE documento_devolucion OWNER TO postgres;
ALTER TABLE documento_devolucion ALTER COLUMN tramite SET STATISTICS 0;
ALTER TABLE documento_devolucion ALTER COLUMN id_documento_faltante SET
STATISTICS 0;

CREATE TABLE documento_pregunta(
    id serial NOT NULL,
    pregunta character varying,
    documento character varying,
    clas character varying,
    seccion integer,
    CONSTRAINT pk_documento_pregunta_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE documento_pregunta OWNER TO postgres;

CREATE TABLE documentos(
    id_documento integer NOT NULL DEFAULT
nextval('documentos_id_documento_seq1'::regclass),
    cod_documento character varying,
    descripcion character varying,
    validaciones text,
    obligatorio integer,
    tipo_documento character varying,
    tipo_gestion character varying,
    casuistica text,
    CONSTRAINT pk_documentos_id_documento_new PRIMARY KEY (id_documento)
)WITH ( OIDS=FALSE);
ALTER TABLE documentos OWNER TO postgres;

```

```

CREATE TABLE documentos_faltantes(
    id_documento_faltante serial NOT NULL,
    nombre_documento character varying NOT NULL,
    activo integer NOT NULL DEFAULT 1,
    CONSTRAINT documentos_faltantes_pkey PRIMARY KEY
(id_documento_faltante)
)WITH ( OIDS=FALSE);
ALTER TABLE documentos_faltantes OWNER TO postgres;
ALTER TABLE documentos_faltantes ALTER COLUMN id_documento_faltante SET
STATISTICS 0;
ALTER TABLE documentos_faltantes ALTER COLUMN nombre_documento SET
STATISTICS 0;
ALTER TABLE documentos_faltantes ALTER COLUMN activo SET STATISTICS 0;

CREATE TABLE documentos_obligatorios(
    na serial NOT NULL,
    clas character varying,
    documento character varying,
    seccion integer,
    CONSTRAINT pk_documentos_obligatorios_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE documentos_obligatorios OWNER TO postgres;

CREATE TABLE eliminamotivotmp(
    consecutivo serial NOT NULL,
    id_mot integer,
    user_insert character varying,
    tramite character varying,
    CONSTRAINT eliminamotivotmp_pkey PRIMARY KEY (consecutivo)
)WITH ( OIDS=FALSE);
ALTER TABLE eliminamotivotmp OWNER TO postgres;

CREATE TABLE entidades_dictamenes
(
    cod_entidad character varying NOT NULL,
    nombre_entidad character varying,
    estado integer DEFAULT 0,
    CONSTRAINT pk_entidades_dictamenes_cod_entidad PRIMARY KEY
(cod_entidad)
)WITH ( OIDS=FALSE);
ALTER TABLE entidades_dictamenes OWNER TO postgres;

CREATE TABLE entidades_gestion(
    na serial NOT NULL,
    tramite character varying,
    nit character varying,
    razon_social character varying,
    estado integer,
    observacion text,
    user_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_entidades_gestion_na PRIMARY KEY (na)
)

```

```

WITH (OIDS=FALSE);
ALTER TABLE entidades_gestion OWNER TO postgres;

CREATE TABLE entidades_prerradicacion(
    id serial NOT NULL,
    cod_prerradicado character varying,
    tipo_historia character varying,
    motivo integer,
    entidad character varying,
    user_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    fecha_desde character varying,
    fecha_hasta character varying,
    salario_real character varying,
    cotiza_iss character varying,
    cotiza_otros character varying,
    CONSTRAINT pk_entidades_prerradicacion PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE entidades_prerradicacion OWNER TO postgres;

CREATE TABLE entidades_prerradicacion_tmp(
    id integer NOT NULL DEFAULT
nextval('entidades_prerradicacion_tmp_id_seq'::regclass),
    cod_prerradicado character varying,
    tipo_historia character varying,
    motivo integer,
    entidad character varying,
    user_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    fecha_desde character varying,
    fecha_hasta character varying,
    salario_real character varying,
    cotiza_iss character varying,
    cotiza_otros character varying,
    CONSTRAINT entidades_prerradicacion_tmp_pk_entidades_prerradicacion
PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE entidades_prerradicacion_tmp OWNER TO postgres;

CREATE TABLE entidades_prerradicacion_tmp(
    id serial NOT NULL,
    guid character varying,
    tipo_historia character varying,
    motivo integer,
    entidad character varying,
    user_insert character varying,
    fecha_desde character varying,
    fecha_hasta character varying,
    salario_real character varying,
    cotiza_iss character varying,
    cotiza_otros character varying,
    CONSTRAINT pk_entidades_prerradicacion_tmp PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE entidades_prerradicacion_tmp OWNER TO postgres;

```

```

CREATE TABLE entidades_tramite(
    na serial NOT NULL,
    tramite character varying,
    nit character varying,
    razon_social character varying,
    estado integer,
    observacion text,
    user_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_entidades_tramite_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE entidades_tramite OWNER TO postgres;

CREATE TABLE envios_documentales (
    id_envio serial NOT NULL,
    cod_prerradicado character varying NOT NULL,
    num_folios integer,
    oficina character varying,
    usuario character varying,
    fecha_envio timestamp without time zone DEFAULT now(),
    estado character varying,
    usuario_imagine character varying,
    fecha_recepcion timestamp without time zone, -- Fecha en la que recibe
    imagine
    folios_recibidos integer,
    CONSTRAINT pk_enviodocumentos PRIMARY KEY (cod_prerradicado)
)WITH ( OIDS=FALSE);
ALTER TABLE envios_documentales OWNER TO postgres;
COMMENT ON COLUMN envios_documentales.fecha_recepcion IS 'Fecha en la que
recibe imagine';

CREATE TABLE eps(
    codigo character varying NOT NULL,
    nit character varying,
    nombre character varying,
    estado integer,
    CONSTRAINT pk_eps_codigo PRIMARY KEY (codigo)
)WITH ( OIDS=FALSE);
ALTER TABLE eps OWNER TO postgres;

CREATE TABLE errores_comparacion(
    na serial NOT NULL,
    tramite character varying,
    error character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    CONSTRAINT pk_errores_comparacion_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE errores_comparacion OWNER TO postgres;

```

```

CREATE TABLE errores_generan_tarea(
    error character varying NOT NULL,
    area_responsable character varying,
    CONSTRAINT pk_errores_generan_tarea_error PRIMARY KEY (error)
)WITH ( OIDS=FALSE);
ALTER TABLE errores_generan_tarea OWNER TO postgres;

CREATE TABLE estados_analisis(
    na serial NOT NULL,
    descripcion character varying,
    bandeja character varying,
    cierra_tarea character varying,
    cambia_bandeja character varying,
    CONSTRAINT pk_estados_analisis_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE estados_analisis OWNER TO postgres;

CREATE TABLE estados_bonos(
    na serial NOT NULL,
    descripcion character varying,
    bandeja character varying,
    origen character varying,
    cierra_tarea character varying,
    cambia_bandeja character varying,
    CONSTRAINT pk_estados_bonos_na PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE estados_bonos OWNER TO postgres;

CREATE TABLE estados_imagine(
    na serial NOT NULL,
    descripcion character varying,
    bandeja character varying,
    cierra_tarea character varying,
    cambia_bandeja character varying,
    CONSTRAINT estados_imagine_pkey PRIMARY KEY (na)
)WITH ( OIDS=FALSE);
ALTER TABLE estados_imagine OWNER TO postgres;
ALTER TABLE estados_imagine ALTER COLUMN na SET STATISTICS 0;
ALTER TABLE estados_imagine ALTER COLUMN descripcion SET STATISTICS 0;
ALTER TABLE estados_imagine ALTER COLUMN bandeja SET STATISTICS 0;
ALTER TABLE estados_imagine ALTER COLUMN cierra_tarea SET STATISTICS 0;
ALTER TABLE estados_imagine ALTER COLUMN cambia_bandeja SET STATISTICS 0;

CREATE TABLE estructura_cierres_masivos(
    id serial NOT NULL,
    bandeja character varying,
    nombre_campo character varying,
    lista_valor character varying,
    descripcion_valor character varying,
    orden_campo integer,
    CONSTRAINT pk_estructura_cierres_masivos_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE estructura_cierres_masivos OWNER TO postgres;

```

```

CREATE TABLE fechas_vencimiento(
    id serial NOT NULL,
    tipo_solicitud character varying,
    dias integer,
    tipo_plazo character varying,
    CONSTRAINT pk_fechas_vencimiento_id PRIMARY KEY (id)
)WITH ( OIDS=FALSE);
ALTER TABLE fechas_vencimiento OWNER TO postgres;

CREATE TABLE festivos(
    dia character varying NOT NULL,
    tipo character(1),
    CONSTRAINT pk_festivos_dia PRIMARY KEY (dia)
)WITH ( OIDS=FALSE);
ALTER TABLE festivos OWNER TO postgres;
GRANT ALL ON TABLE festivos TO postgres;
CREATE INDEX idx_festivos_dia ON festivos
    USING btree (dia COLLATE pg_catalog."default");

CREATE INDEX idx_pensionados_festivos_dia ON festivos
    USING btree (dia COLLATE pg_catalog."default");

CREATE TABLE gestion_analisis(
    id_gestion serial NOT NULL,
    tramite character varying,
    estado integer,
    observacion text,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    accion_legal character varying,
    pago_ant_bono character varying,
    pago_ant_cobro character varying,
    notificado_por character varying,
    CONSTRAINT pk_gestion_analisis_id_gestion PRIMARY KEY (id_gestion)
)WITH ( OIDS=FALSE);
ALTER TABLE gestion_analisis OWNER TO postgres;

-- Index: idx_gestion_analisis_fecha_insert

-- DROP INDEX idx_gestion_analisis_fecha_insert;

CREATE INDEX idx_gestion_analisis_fecha_insert
    ON gestion_analisis
    USING btree
    (fecha_insert);

-- Index: idx_gestion_analisis_tramite

-- DROP INDEX idx_gestion_analisis_tramite;

CREATE INDEX idx_gestion_analisis_tramite
    ON gestion_analisis
    USING btree
    (tramite COLLATE pg_catalog."default");

```

```

-- Index: idx_gestion_analisis_usuario_insert

-- DROP INDEX idx_gestion_analisis_usuario_insert;

CREATE INDEX idx_gestion_analisis_usuario_insert
  ON gestion_analisis
  USING btree
  (usuario_insert COLLATE pg_catalog."default");

CREATE TABLE gestion_bandejas(
  id_gestion serial NOT NULL,
  tramite character varying,
  estado_gestion integer,
  observacion text,
  user_gestion character varying,
  fecha_gestion timestamp without time zone DEFAULT now(),
  origen_gestion character varying,
  sub_estado character varying,
  entidad integer,
  CONSTRAINT pk_gestion_bandejas_id_gestion PRIMARY KEY (id_gestion)
)WITH ( OIDS=FALSE);
ALTER TABLE gestion_bandejas OWNER TO postgres;

CREATE TABLE gestion_bonos
(
  id_gestion serial NOT NULL,
  tramite character varying,
  estado_gestion integer,
  observacion text,
  aplica_ent character varying,
  id_entidad integer,
  user_gestion character varying,
  fecha_gestion timestamp without time zone DEFAULT now(),
  origen_gestion character varying,
  CONSTRAINT pk_gestion_bonos_id_gestion PRIMARY KEY (id_gestion)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE gestion_bonos
  OWNER TO postgres;

CREATE TABLE gestion_codificaciones
(
  consecutivo serial NOT NULL,
  codificacion_actual character varying,
  codificacion_futura character varying,
  bandeja character varying,
  CONSTRAINT gestion_codificaciones_pkey PRIMARY KEY (consecutivo)
)
WITH (
  OIDS=FALSE

```



```

);
ALTER TABLE gestion_codificaciones
  OWNER TO postgres;

CREATE TABLE gestion_conceptos
(
  id_gestion serial NOT NULL,
  contacta character varying,
  fecha_problabre_rad character varying,
  fallecido character varying,
  observacion text,
  usuario_gestion character varying,
  fecha_gestion timestamp without time zone DEFAULT now(),
  id_concepto integer,
  area_gestion character varying DEFAULT 'Call Center'::character
varying,
  origen_gestion character varying,
  cod_futura integer,
  CONSTRAINT pk_gestion_conceptos_id_gestion PRIMARY KEY (id_gestion),
  CONSTRAINT fk_gestion_conceptos_id_concepto FOREIGN KEY (id_concepto)
    REFERENCES conceptos_medicos (id_concepto) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE gestion_conceptos
  OWNER TO postgres;

CREATE TABLE gestion_dictamenes_otrasent
(
  id serial NOT NULL,
  id_dictamen integer,
  fecha_insert timestamp without time zone DEFAULT now(),
  usuario character varying,
  observacion character varying,
  estado character varying
)
WITH (
  OIDS=FALSE
);
ALTER TABLE gestion_dictamenes_otrasent
  OWNER TO postgres;

CREATE TABLE gestion_funeraria
(
  id_gestion serial NOT NULL,
  id_reclamante integer,
  tramite character varying,
  estado character varying,
  observacion text,
  contacto_gestion character varying,
  reclama_paga character varying,
  poder character varying,

```

```

pagado_por character varying,
usuario_gestion character varying,
fecha_gestion timestamp without time zone DEFAULT now(),
plan_exequial character varying,
numero_plan character varying,
titular_fallecido character varying,
nombre_funeraria character varying,
estado_cruce character varying,
CONSTRAINT pk_gestion_funeraria_id_gestion PRIMARY KEY (id_gestion)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE gestion_funeraria
    OWNER TO postgres;

CREATE TABLE gestion_imagine
(
    id_gestion serial NOT NULL,
    tramite character varying,
    estado integer,
    observacion text,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    CONSTRAINT gestion_imagine_pkey PRIMARY KEY (id_gestion)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE gestion_imagine
    OWNER TO postgres;
ALTER TABLE gestion_imagine ALTER COLUMN id_gestion SET STATISTICS 0;
ALTER TABLE gestion_imagine ALTER COLUMN tramite SET STATISTICS 0;
ALTER TABLE gestion_imagine ALTER COLUMN estado SET STATISTICS 0;
ALTER TABLE gestion_imagine ALTER COLUMN observacion SET STATISTICS 0;
ALTER TABLE gestion_imagine ALTER COLUMN fecha_insert SET STATISTICS 0;
ALTER TABLE gestion_imagine ALTER COLUMN usuario_insert SET STATISTICS 0;

CREATE TABLE gestion_it
(
    id_gestion serial NOT NULL,
    tramite character varying,
    estado character varying,
    observacion text,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    ipp integer,
    CONSTRAINT pk_gestion_it_id_gestion PRIMARY KEY (id_gestion)
)
WITH (
    OIDS=FALSE
);

```

```

ALTER TABLE gestion_it
  OWNER TO postgres;

CREATE TABLE gestion_prerradicacion
(
  id_gestion serial NOT NULL,
  tramite character varying NOT NULL,
  contacta character varying,
  observacion text,
  usuario_gestion character varying,
  fecha_gestion timestamp without time zone DEFAULT now(),
  tipo_gestion character varying,
  cod_futura integer,
  fecha_agenda character varying,
  CONSTRAINT pk_gestion_prerradicacion_id_gestion PRIMARY KEY
(id_gestion)
);
WITH (
  OIDS=FALSE
);
ALTER TABLE gestion_prerradicacion
  OWNER TO postgres;

CREATE TABLE gestion_prerrequisitos
(
  na_gestion serial NOT NULL,
  tramite character varying,
  area character varying,
  tarea character varying,
  respuesta_tarea character varying,
  observacion text,
  fecha_insert timestamp without time zone DEFAULT now(),
  usuario_insert character varying,
  genera_tarea character varying,
  tipopro character varying,
  emisor character varying,
  fecha_estructuracion character varying,
  porcentaje_pcl character varying,
  etapa character varying DEFAULT 'Prerradicacion'::character varying,
  CONSTRAINT pk_gestion_prerrequisitos_na_gestion PRIMARY KEY
(na_gestion)
);
WITH (
  OIDS=FALSE
);
ALTER TABLE gestion_prerrequisitos
  OWNER TO postgres;

-- Index: idx_gestion_prerrequisitos_tramite

-- DROP INDEX idx_gestion_prerrequisitos_tramite;

CREATE INDEX idx_gestion_prerrequisitos_tramite
  ON gestion_prerrequisitos

```

```

    USING btree
    (tramite COLLATE pg_catalog."default");

CREATE TABLE gestion_solicitudes_ips
(
    id_gestion serial NOT NULL,
    tramite character varying,
    devolucion bigint,
    estado_gestion character varying,
    contacto character varying,
    telefono double precision,
    observacion text,
    fecha_insert timestamp without time zone,
    usuario_insert character varying,
    email character varying,
    direccion character varying,
    celular double precision,
    CONSTRAINT pk_gestion_solicitudes_ips_id_gestion PRIMARY KEY
(id_gestion)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE gestion_solicitudes_ips
    OWNER TO postgres;

CREATE TABLE historico_asignacion
(
    id serial NOT NULL,
    tramite character varying,
    usuario character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    bandeja character varying,
    CONSTRAINT pk_historico_asignacion_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE historico_asignacion
    OWNER TO postgres;

CREATE TABLE historico_reasignacion
(
    id serial NOT NULL,
    tramite character varying,
    usuario_asignado character varying,
    usuario_asigna character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_historico_reasignacion_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE historico_reasignacion

```

```

OWNER TO postgres;

CREATE TABLE homologacion_estados
(
    id serial NOT NULL,
    estado_principal character varying,
    sub_estado character varying,
    sub_estado_traza character varying,
    estado_tarea character varying,
    CONSTRAINT pk_homologacion_estados_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE homologacion_estados
OWNER TO postgres;

CREATE TABLE imagenes_prerradicacion
(
    id serial NOT NULL ,
    cod_prerradicado character varying,
    path character varying,
    user_insert character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    origen_imagen character varying,
    CONSTRAINT pk_adjuntos_prerradicacion PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE imagenes_prerradicacion
OWNER TO postgres;

CREATE TABLE indicador_errores
(
    na serial NOT NULL,
    bandeja_origen character varying,
    bandeja_destino character varying,
    tramite character varying,
    rechazo character varying,
    obs_rechazo text,
    fecha_rechazo timestamp without time zone DEFAULT now(),
    usuario_rechaza character varying,
    CONSTRAINT pk_indicador_errores_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE indicador_errores
OWNER TO postgres;

CREATE TABLE info_pres_economica_afil

```

```

(
  na serial NOT NULL,
  id_afiliado character varying,
  tipo_doc character varying,
  papell character varying,
  sapell character varying,
  pnomb character varying,
  snomb character varying,
  fecha_nacimiento character varying(10),
  fecha_siniestro character varying(10),
  tipo_solicitud character varying,
  genero character(1), -- M = Masculino...
  fecha_solicitud character varying(10),
  numero_beneficiarios integer,
  fecha_insert timestamp without time zone DEFAULT now(),
  user_insert character varying,
  estado character varying DEFAULT 'Ingresada'::character varying,
  tramite character varying,
  cod_oficina character varying,
  estado_civil character varying,
  origen_siniestro character varying,
  causal_siniestro character varying,
  email character varying,
  nom_solicitante character varying,
  direccion_solicitante character varying,
  telefono_solicitante character varying,
  ciudad_depto_solicitante character varying,
  CONSTRAINT pk_info_pres_economica_afil_na PRIMARY KEY (na)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE info_pres_economica_afil
  OWNER TO postgres;
GRANT ALL ON TABLE info_pres_economica_afil TO postgres;
COMMENT ON COLUMN info_pres_economica_afil.genero IS 'M = Masculino
F = Femenino';

-- Index: idx_info_pres_economica_afil_fecha_solicitud

-- DROP INDEX idx_info_pres_economica_afil_fecha_solicitud;

CREATE INDEX idx_info_pres_economica_afil_fecha_solicitud
  ON info_pres_economica_afil
  USING btree
  (fecha_solicitud COLLATE pg_catalog."default");

CREATE TABLE info_pres_economica_afil_sp
(
  na serial NOT NULL,
  id_afiliado character varying,
  tipo_doc character varying,

```

```

papell character varying,
sapell character varying,
pnomb character varying,
snomb character varying,
fecha_nacimiento character varying(10),
fecha_siniestro character varying(10),
tipo_solicitud character varying,
genero character(1), -- M = Masculino...
fecha_solicitud character varying(10),
numero_beneficiarios integer,
fecha_insert timestamp without time zone DEFAULT now(),
user_insert character varying,
estado character varying DEFAULT 'Ingresada'::character varying,
estado_civil character varying,
origen_siniestro character varying,
causal_siniestro character varying,
email character varying,
tramite character varying,
CONSTRAINT pk_info_pres_economica_afil_sp_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE info_pres_economica_afil_sp
    OWNER TO postgres;
GRANT ALL ON TABLE info_pres_economica_afil_sp TO postgres;
COMMENT ON COLUMN info_pres_economica_afil_sp.genero IS 'M = Masculino
F = Femenino';

-- Index: idx_info_pres_economica_afil_sp_fecha_solicitud
-- DROP INDEX idx_info_pres_economica_afil_sp_fecha_solicitud;
CREATE INDEX idx_info_pres_economica_afil_sp_fecha_solicitud
    ON info_pres_economica_afil_sp
    USING btree
    (fecha_solicitud COLLATE pg_catalog."default");

-- Index: idx_info_pres_economica_afil_sp_id_afiliado
-- DROP INDEX idx_info_pres_economica_afil_sp_id_afiliado;
CREATE INDEX idx_info_pres_economica_afil_sp_id_afiliado
    ON info_pres_economica_afil_sp
    USING btree
    (id_afiliado COLLATE pg_catalog."default");

-- Index: idx_info_pres_economica_afil_sp_tipo_solicitud
-- DROP INDEX idx_info_pres_economica_afil_sp_tipo_solicitud;
CREATE INDEX idx_info_pres_economica_afil_sp_tipo_solicitud
    ON info_pres_economica_afil_sp

```

```

        USING btree
        (tipo_solicitud COLLATE pg_catalog."default");

CREATE TABLE info_prestacion_economica_ben
(
    na_beneficiario serial NOT NULL,
    na_afiliado double precision,
    id_beneficiario character varying,
    tipo_doc_ben character varying,
    papell_ben character varying,
    sapell_ben character varying,
    pnomb_ben character varying,
    snomb_ben character varying,
    fecha_nacimiento_ben character varying(10),
    fecha_sol_ben character varying(10),
    genero_ben character(1), -- M = Masculino...
    parentesco_ben integer,
    calidad_ben character varying,
    fecha_insert_ben timestamp without time zone DEFAULT now(),
    user_insert character varying,
    CONSTRAINT pk_info_prestacion_economica_ben_na_beneficiario PRIMARY KEY
    (na_beneficiario)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE info_prestacion_economica_ben
    OWNER TO postgres;
GRANT ALL ON TABLE info_prestacion_economica_ben TO postgres;

COMMENT ON COLUMN info_prestacion_economica_ben.genero_ben IS 'M =
Masculino
F = Femenino';

-- Index: idx_info_prestacion_economica_ben_na_afiliado
-- DROP INDEX idx_info_prestacion_economica_ben_na_afiliado;

CREATE INDEX idx_info_prestacion_economica_ben_na_afiliado
    ON info_prestacion_economica_ben
    USING btree
    (na_afiliado);

CREATE TABLE info_prestacion_economica_ben_sp
(
    na_beneficiario serial NOT NULL,
    na_afiliado double precision,
    id_beneficiario character varying,
    tipo_doc_ben character varying,
    papell_ben character varying,
    sapell_ben character varying,

```



```

    pnomb_ben character varying,
    snomb_ben character varying,
    fecha_nacimiento_ben character varying(10),
    fecha_sol_ben character varying(10),
    genero_ben character(1), -- M = Masculino...
    parentesco_ben integer,
    calidad_ben character varying,
    fecha_insert_ben timestamp without time zone DEFAULT now(),
    user_insert character varying,
    CONSTRAINT pk_info_prestacion_economica_ben_sp_na_beneficiario PRIMARY
KEY (na_beneficiario)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE info_prestacion_economica_ben_sp
    OWNER TO postgres;
GRANT ALL ON TABLE info_prestacion_economica_ben_sp TO postgres;

COMMENT ON COLUMN info_prestacion_economica_ben_sp.genero_ben IS 'M =
Masculino
F = Femenino';

CREATE TABLE log_cambio_fecharad
(
    sr serial NOT NULL,
    fechainser timestamp without time zone DEFAULT now(),
    tramite character varying,
    fecha_ant character varying,
    fecha_act character varying,
    usuario_mod character varying
)
WITH (
    OIDS=FALSE
);
ALTER TABLE log_cambio_fecharad
    OWNER TO postgres;

CREATE TABLE log_edicion
(
    id_edicion serial NOT NULL,
    tramite character varying,
    user_edita character varying,
    fecha_edita timestamp without time zone DEFAULT now(),
    CONSTRAINT log_edicion_pkey PRIMARY KEY (id_edicion)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE log_edicion
    OWNER TO postgres;

CREATE TABLE log_eliminacion
(

```

```

consecutivo serial NOT NULL,
tramite character varying,
motivo character varying,
usuario_insert character varying,
fecha_insert timestamp without time zone DEFAULT now(),
CONSTRAINT log_eliminacion_pkey PRIMARY KEY (consecutivo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE log_eliminacion
    OWNER TO postgres;

CREATE TABLE log_pasos_analisis
(
    na serial NOT NULL,
    tramite character varying,
    usuario character varying,
    fecha_modificacion timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_log_pasos_analisis_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE log_pasos_analisis
    OWNER TO postgres;

CREATE TABLE log_prerradicacion
(
    id integer NOT NULL,
    user_insert character varying,
    user_actualiza character varying,
    CONSTRAINT log_prerradicacion_pkey PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE log_prerradicacion
    OWNER TO postgres;

CREATE TABLE matching_prerrequisitos
(
    na serial NOT NULL,
    tramite character varying,
    primer_nombre character varying,
    segundo_nombre character varying,
    primer_apellido character varying,
    segundo_apellido character varying,
    fecha_nacimiento character varying,
    usuario_inserta character varying,
    tipo_proceso character varying,
    genero character varying,
    fecha_gestion timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_matching_prerrequisitos_na PRIMARY KEY (na)
)

```

```

)
WITH (
    OIDS=FALSE
);
ALTER TABLE matching_prerrequisitos
    OWNER TO postgres;

CREATE TABLE matching_prerrequisitos_tmp
(
    na serial NOT NULL,
    tramite character varying,
    primer_nombre character varying,
    segundo_nombre character varying,
    primer_apellido character varying,
    segundo_apellido character varying,
    fecha_nacimiento character varying,
    usuario_inserta character varying,
    tipo_proceso character varying,
    genero character varying,
    CONSTRAINT pk_matching_prerrequisitos_tmp_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE matching_prerrequisitos_tmp
    OWNER TO postgres;

CREATE TABLE mensajes_obp
(
    id serial NOT NULL,
    mensaje text,
    estado integer DEFAULT 1,
    cod_mensaje character varying,
    CONSTRAINT pk_mensajes_obp_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE mensajes_obp
    OWNER TO postgres;

CREATE TABLE mensajes_obp_tramite
(
    na serial NOT NULL,
    tramite character varying,
    id_mensaje integer,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    CONSTRAINT pk_mensajes_obp_tramite_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);

```

```

ALTER TABLE mensajes_obp_tramite
  OWNER TO postgres;

CREATE TABLE menu
(
  id serial NOT NULL,
  status character varying,
  padre integer,
  nombre character varying,
  icono character varying,
  url character varying,
  orden integer,
  padre_ppal integer,
  CONSTRAINT pk_menu_id PRIMARY KEY (id)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE menu
  OWNER TO postgres;

CREATE TABLE motivos_prerradicacion
(
  id serial NOT NULL,
  descripcion character varying
)
WITH (
  OIDS=FALSE
);
ALTER TABLE motivos_prerradicacion
  OWNER TO postgres;

CREATE TABLE notificaciones
(
  na serial NOT NULL,
  tramite character varying,
  tipo_notificacion character varying,
  fecha_envio timestamp without time zone DEFAULT now(),
  usuario_envio character varying,
  mail_envio character varying,
  path character varying,
  CONSTRAINT pk_notificaciones_na PRIMARY KEY (na)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE notificaciones
  OWNER TO postgres;

CREATE TABLE oferta_servicio
(
  na serial NOT NULL,
  tipo_solicitud character varying,
  oferta_mesa integer,

```

```

oferta_prerrequisitos integer,
oferta_prerradicacion integer,
oferta_precategorizacion integer,
plazo_legal integer,
CONSTRAINT pk_oferta_servicio_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE oferta_servicio
    OWNER TO postgres;

CREATE TABLE ofertas_bandejas
(
    id_oferta serial NOT NULL,
    bandeja character varying,
    tiposol character varying,
    tipo_oferta character varying,
    tiempo_dias integer,
    asignado_por character varying,
    fecha_asignacion_oferta timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_ofertas_bandejas_id_oferta PRIMARY KEY (id_oferta)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE ofertas_bandejas
    OWNER TO postgres;

CREATE TABLE oficinas
(
    codigo character varying NOT NULL,
    titulo character varying,
    regional_cod character varying,
    regional character varying,
    CONSTRAINT pk_oficinas_codigo PRIMARY KEY (codigo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE oficinas
    OWNER TO postgres;

CREATE TABLE panel_asignacion
(
    id serial NOT NULL,
    usuario character varying,
    tiposol character varying,
    regional character varying,
    sub_estado character varying,
    origen character varying,
    estado integer,

```

```

    responsable character varying,
    CONSTRAINT pk_panel_asignacion_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE panel_asignacion
    OWNER TO postgres;

CREATE TABLE parentesco
(
    id_parentesco serial NOT NULL,
    parentesco character varying,
    CONSTRAINT pk_parentesco_id_parentesco PRIMARY KEY (id_parentesco)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE parentesco
    OWNER TO postgres;
GRANT ALL ON TABLE parentesco TO postgres;

CREATE TABLE permisos_menu
(
    id serial NOT NULL,
    usuario character varying,
    menu integer,
    CONSTRAINT pk_permisos_menu_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE permisos_menu
    OWNER TO postgres;

CREATE TABLE prerradicacion
(
    id character varying NOT NULL,
    tipodoc character varying,
    telefono character varying,
    celular bigint,
    autorizasms character varying,
    autoriza_envio character varying,
    ciudad character varying,
    tiposol character varying,
    his_lab_bono character varying,
    mot_his_lab_bono text,
    his_lab_prot character varying,
    mot_his_lab_prot text,
    his_lab_otras character varying,
    mot_his_lab_otras text,
    usuario_insert character varying,

```

```

fecha_insert timestamp without time zone DEFAULT now(),
estado character varying,
correo character varying,
prima_media character varying,
enti_publica_desde character varying,
enti_publica_hasta character varying,
subtipo integer,
primer_nombre character varying,
segundo_nombre character varying,
primer_apellido character varying,
segundo_apellido character varying,
fecha_vencimiento character varying,
ent_histlabbono integer,
ent_histlabprot integer,
ent_histlabotras integer,
ind_normalizacion character(1),
ent_normalizacion character varying,
derecho_bono character(1),
tramite_bono character(1),
entidad_contribuyente character varying,
investigacion character(1),
categoria character varying DEFAULT 'PREAPROBADO'::character varying,
estado_documentos character varying NOT NULL DEFAULT 'Cargar
Documentos'::character varying,
direccion_corresp character varying,
dias_incapacitado integer DEFAULT 0,
tipo_vinculacion character varying,
rec_apoderado character varying,
nom_apoderado character varying,
numero_grupo integer DEFAULT 1,
saldo character varying,
folofc integer,
foleps integer,
producto character varying,
oficina character varying,
estado_radificacion character varying DEFAULT 'PENDIENTE
RADICAR'::character varying,
user_radica character varying,
fecha_radica timestamp without time zone,
estado_general character varying,
documentos_oficina character varying,
documentos_ips character varying,
observacion_ips text,
categoria_real character varying,
responsable character varying,
fecha_asignacion timestamp without time zone,
vencimiento_legal character varying,
estado_analisis integer DEFAULT 1,
cedula character varying,
tipo_registro character varying DEFAULT 'Dia a Dia'::character varying,
fecha_radificacion_sistema timestamp without time zone,
carga_analisis character varying,
tipo_visita character varying,
estado_marca character varying,

```

```

tramite_reactiva character varying,
marca_tutela character varying,
observacion_tutela text,
fecha_marca_tutela timestamp without time zone,
usuario_marca_tutela character varying,
fecha_marca_estado timestamp without time zone,
observacion_estado text,
usuario_marca_estado character varying,
estado_grupo character varying,
marca_grupos character varying DEFAULT 'N'::character varying,
bono_post_analisis character varying,
analisis_post_bono character varying,
caso_reabierto character varying,
fecha_reapertura timestamp without time zone,
observacion_reapertura text,
usuario_reapertura character varying,
analisis_doc_pend character varying DEFAULT 'N'::character varying,
ind_post_analisis_grupo character varying DEFAULT 'N'::character
varying,
grupo_activo character varying DEFAULT 'N'::character varying,
CONSTRAINT pkey_prerradicacion PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE prerradicacion
    OWNER TO postgres;

-- Index: idx_prerradicacion_cedula

-- DROP INDEX idx_prerradicacion_cedula;

CREATE INDEX idx_prerradicacion_cedula
    ON prerradicacion
    USING btree
    (cedula COLLATE pg_catalog."default");

-- Index: idx_prerradicacion_tiposol

-- DROP INDEX idx_prerradicacion_tiposol;

CREATE INDEX idx_prerradicacion_tiposol
    ON prerradicacion
    USING btree
    (tiposol COLLATE pg_catalog."default");

CREATE TABLE prerradicacion_beneficiarios
(
    id serial NOT NULL,
    codigo_prerradicacion character varying,
    beneficiario integer,
    CONSTRAINT pk_prerad_benef PRIMARY KEY (id)
)

```



```

WITH (
    OIDS=FALSE
);
ALTER TABLE prerradicacion_beneficiarios
    OWNER TO postgres;

CREATE TABLE prerequisites_automaticos
(
    na serial NOT NULL,
    id_afiliado character varying,
    tipo_solicitud character varying,
    rezagos character varying,
    obs_rezagos character varying,
    planillas character varying,
    observacion_planillas character varying,
    depositos character varying,
    observacion_depositos character varying,
    deudas character varying,
    observacion_deudas character varying,
    estado character varying DEFAULT 'Cargado'::character varying,
    CONSTRAINT pk_prerequisites_automaticos_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE prerequisites_automaticos
    OWNER TO postgres;

CREATE TABLE producto
(
    codigo character varying NOT NULL,
    descripcion character varying,
    estado integer DEFAULT 0,
    CONSTRAINT pk_producto_codigo PRIMARY KEY (codigo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE producto
    OWNER TO postgres;

CREATE TABLE punteo_solicitudes
(
    na serial NOT NULL,
    tramite character varying,
    folios_recibidos integer,
    usuario_recibe character varying,
    fecha_punteo timestamp without time zone DEFAULT now(),
    estado_punteo character varying,
    numero_grupo integer,
    sub_estado character varying,
    CONSTRAINT pk_punteo_solicitudes_na PRIMARY KEY (na)
)
WITH (

```

```

        OIDS=FALSE
    );
    ALTER TABLE punteo_solicitudes
        OWNER TO postgres;
    GRANT ALL ON TABLE punteo_solicitudes TO postgres;

CREATE TABLE reagendamiento
(
    id serial NOT NULL,
    tramite character varying,
    fecha_inicial_agenda timestamp without time zone,
    oficina character varying,
    asesor character varying,
    fecha_solicita_reagendar timestamp without time zone DEFAULT now(),
    usuario_solicita_reagendar character varying,
    CONSTRAINT pk_reagendamiento_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE reagendamiento
    OWNER TO postgres;

CREATE TABLE recepcion_documentos
(
    na serial NOT NULL,
    na_documento double precision,
    tipo_documento character varying,
    usuario_recibe character varying,
    fecha_recibe timestamp without time zone DEFAULT now(),
    usuario_recibe_imagine character varying,
    fecha_recibe_imagine timestamp without time zone,
    estado_recibe character varying,
    estado_recibe_imagine character varying,
    tramite character varying,
    id_afiliado character varying,
    tipo_solicitud character varying,
    CONSTRAINT pk_recepcion_documentos_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE recepcion_documentos
    OWNER TO postgres;

CREATE TABLE recepcion_documentos_ips
(
    id serial NOT NULL,
    na_documento integer,
    usuario_recibe_oficina character varying,
    fecha_recibe_oficina timestamp without time zone DEFAULT now(),
    usuario_recibe_ips character varying,
    fecha_recibe_ips timestamp without time zone,

```

```

    estado_recepcion character varying,
    observacion_ips text,
    CONSTRAINT pk_recepcion_documentos_ips_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE recepcion_documentos_ips
    OWNER TO postgres;

CREATE TABLE resp_cuestionario
(
    cod_prerradicado character varying,
    cod_pregunta character varying,
    respuesta character varying,
    CONSTRAINT fk_resp_cuestionario FOREIGN KEY (cod_prerradicado)
        REFERENCES prerradicacion (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
    OIDS=FALSE
);
ALTER TABLE resp_cuestionario
    OWNER TO postgres;

CREATE TABLE responsables
(
    id_responsable serial NOT NULL,
    usuario character varying,
    tipo_solicitud character varying,
    categoria character varying,
    bandeja character varying,
    activo integer DEFAULT 1,
    CONSTRAINT pk_responsables_id_responsable PRIMARY KEY (id_responsable)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE responsables
    OWNER TO postgres;

CREATE TABLE responsables_bonos
(
    id_responsable serial NOT NULL,
    usuario character varying,
    regional character varying,
    estado integer DEFAULT 1,
    CONSTRAINT pk_responsables_bonos_id_responsable PRIMARY KEY
(id_responsable)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE responsables_bonos

```

```

OWNER TO postgres;

CREATE TABLE responsables_imagine
(
    id_resp serial NOT NULL,
    usuario character varying,
    sede character varying,
    bandeja character varying,
    estado integer,
    CONSTRAINT pk_responsables_imagine PRIMARY KEY (id_resp)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE responsables_imagine
OWNER TO postgres;

CREATE TABLE responsables_tramite
(
    id_responsable serial NOT NULL,
    usuario character varying,
    categoria character varying,
    estado integer DEFAULT 1,
    CONSTRAINT pk_responsables_tramite_id_responsable PRIMARY KEY
(id_responsable)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE responsables_tramite
OWNER TO postgres;

CREATE TABLE secciones_guia
(
    na serial NOT NULL,
    descripcion character varying,
    orden integer,
    CONSTRAINT pk_secciones_guia_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE secciones_guia
OWNER TO postgres;

CREATE TABLE segundos_grupos
(
    na serial NOT NULL,
    tramite character varying,
    estado_grupo character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    numero_grupo integer,
    creado_as character varying,

```

```

    CONSTRAINT pk_segundos_grupos_na PRIMARY KEY (na)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE segundos_grupos
    OWNER TO postgres;

CREATE TABLE solicitudes_rechazadas
(
    consecutivo serial NOT NULL,
    tramite character varying,
    documento character varying,
    causal bigint,
    ccben character varying,
    observacion text,
    usuario_rechazo character varying,
    fecha_rechazo timestamp without time zone DEFAULT now(),
    estado integer DEFAULT 0,
    CONSTRAINT pk_solicitudes_rechazadas_consecutivo PRIMARY KEY
(consecutivo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE solicitudes_rechazadas
    OWNER TO postgres;

CREATE TABLE solicitudes_rechazadas_tmp
(
    consecutivo serial NOT NULL,
    tramite character varying,
    documento character varying,
    causal bigint,
    ccben character varying,
    observacion text,
    usuario_rechazo character varying,
    fecha_rechazo timestamp without time zone DEFAULT now(),
    estado integer DEFAULT 0,
    CONSTRAINT pk_solicitudes_rechazadas_tmp_consecutivo PRIMARY KEY
(consecutivo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE solicitudes_rechazadas_tmp
    OWNER TO postgres;

CREATE TABLE subtipos
(
    id serial NOT NULL,
    codigo character varying,
    tipo character varying,

```

```

        descripcion character varying,
        CONSTRAINT pk_subtipos PRIMARY KEY (id)
    )
    WITH (
        OIDS=FALSE
    );
    ALTER TABLE subtipos
        OWNER TO postgres;

CREATE TABLE tareas_inconsistencias
(
    id_tarea serial NOT NULL,
    nombre character varying,
    proceso_relacionado character varying,
    CONSTRAINT pk_tareas_inconsistencias_id_tarea PRIMARY KEY (id_tarea)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tareas_inconsistencias
    OWNER TO postgres;

CREATE TABLE tarifas
(
    sr serial NOT NULL,
    tipo_solicitud character varying,
    proceso character varying,
    tarifa integer,
    CONSTRAINT pk_tarifas_sr PRIMARY KEY (sr)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tarifas
    OWNER TO postgres;

CREATE TABLE tipos_beneficiario
(
    id serial NOT NULL,
    beneficiario character varying,
    CONSTRAINT pk_tipos_beneficiario_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tipos_beneficiario
    OWNER TO postgres;

CREATE TABLE tipos_cobro
(
    id serial NOT NULL,
    tramite character varying,

```

```

    tipo_cobro character varying,
    CONSTRAINT pk_tipos_cobro_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tipos_cobro
    OWNER TO postgres;

CREATE TABLE tipos_documento
(
    tipo character varying NOT NULL,
    nombre_tipo character varying,
    CONSTRAINT pk_tipos_documento_tipo PRIMARY KEY (tipo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tipos_documento
    OWNER TO postgres;

CREATE TABLE tipos_prestacion
(
    codigo character varying(3) NOT NULL,
    descripcion character varying(20),
    CONSTRAINT pkey_tip_sol PRIMARY KEY (codigo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tipos_prestacion
    OWNER TO postgres;

CREATE TABLE tmp_rechazos
(
    id serial NOT NULL,
    tramite character varying,
    documento character varying,
    causal bigint,
    ccben character varying,
    observacion text,
    usuario_rechazo character varying,
    fecha_rechazo timestamp without time zone DEFAULT now(),
    CONSTRAINT pk_tmp_rechazos_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE tmp_rechazos
    OWNER TO postgres;

CREATE TABLE traza_asignacion

```

```

(
    id_traza serial NOT NULL,
    usuario_asignado character varying,
    usuario_precedente character varying,
    fecha_asignacion timestamp without time zone DEFAULT now(),
    estado_tarea character varying DEFAULT 'Abierta'::character varying,
    tramite character varying,
    actividad character varying,
    fecha_vencimiento character varying,
    CONSTRAINT pk_traza_asignacion_id_traza PRIMARY KEY (id_traza)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE traza_asignacion
    OWNER TO postgres;

CREATE TABLE trazabilidad_conceptos
(
    consecutivo serial NOT NULL,
    concepto character varying,
    proceso character varying,
    estado_actual character varying,
    estado_precedente character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    sub_estado character varying,
    estado_tarea character varying DEFAULT 'Abierta'::character varying,
    usuario_cierre character varying,
    fecha_cierre timestamp without time zone,
    fecha_vencimiento character varying,
    origen character varying,
    CONSTRAINT pk_trazabilidad_conceptos_consecutivo PRIMARY KEY
(consecutivo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE trazabilidad_conceptos
    OWNER TO postgres;
GRANT ALL ON TABLE trazabilidad_conceptos TO postgres;

-- Index: idx_trazabilidad_conceptos_tramite

-- DROP INDEX idx_trazabilidad_conceptos_tramite;

CREATE INDEX idx_trazabilidad_conceptos_tramite
    ON trazabilidad_conceptos
    USING btree
    (concepto COLLATE pg_catalog."default");

CREATE TABLE trazabilidad_prestaciones

```



```

(
    consecutivo serial NOT NULL,
    tramite character varying,
    proceso character varying,
    estado_actual character varying,
    estado_precedente character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    sub_estado character varying,
    estado_tarea character varying DEFAULT 'Abierta'::character varying,
    usuario_cierre character varying,
    fecha_cierre timestamp without time zone,
    fecha_vencimiento character varying,
    origen character varying,
    CONSTRAINT pk_trazabilidad_prestaciones_consecutivo PRIMARY KEY
(consecutivo)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE trazabilidad_prestaciones
    OWNER TO postgres;
GRANT ALL ON TABLE trazabilidad_prestaciones TO postgres;

-- Index: idx_trazabilidad_prestaciones_tramite
-- DROP INDEX idx_trazabilidad_prestaciones_tramite;

CREATE INDEX idx_trazabilidad_prestaciones_tramite
ON trazabilidad_prestaciones
USING btree
(tramite COLLATE pg_catalog."default");

CREATE TABLE trazabilidad_prestaciones_elim
(
    consecutivo bigint NOT NULL,
    tramite character varying,
    proceso character varying,
    estado_actual character varying,
    estado_precedente character varying,
    fecha_insert timestamp without time zone DEFAULT now(),
    usuario_insert character varying,
    sub_estado character varying,
    estado_tarea character varying DEFAULT 'Abierta'::character varying,
    usuario_cierre character varying,
    fecha_cierre timestamp without time zone,
    fecha_vencimiento character varying,
    origen character varying,
    CONSTRAINT pk_trazabilidad_prestaciones_elim_consecutivo PRIMARY KEY
(consecutivo)
)
WITH (

```

```

    OIDS=FALSE
);
ALTER TABLE trazabilidad_prestaciones_elim
    OWNER TO postgres;
GRANT ALL ON TABLE trazabilidad_prestaciones_elim TO postgres;

CREATE TABLE usuarios
(
    id serial NOT NULL,
    usuario character varying,
    nombre character varying,
    area character varying,
    estado integer,
    sede character varying,
    CONSTRAINT pk_usuarios_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE usuarios
    OWNER TO postgres;

CREATE TABLE validacion_punto_revision
(
    id serial NOT NULL,
    id_trazabilidad_prestaciones integer,
    estado integer,
    CONSTRAINT pk_validacion_punto_revision_id PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE validacion_punto_revision
    OWNER TO postgres;

```

5.3.3. Diccionario de Datos

5.3.3.1. Listado de tablas

A continuación enunciamos el listado de tablas del sistema

Tabla 1 Listado general de tablas del sistema

Table	Owner	Comment
acciones_legales	postgres	
acuerdos_servicio	postgres	
adj_conceptos	postgres	
adj_dictámenes	postgres	
asignacion_bonos	postgres	
asignacion_gestion	postgres	
asignacion_imagine	postgres	
asignacion_tramites	postgres	
beneficiarios_prerradicacion	postgres	
beneficiarios_prerradicacion_temp	postgres	
beneficiarios_temp	postgres	
casos_devueltos	postgres	
causales_rechazo	postgres	
cerradas_por_cierre	postgres	
ciudades	postgres	
clasificacion_tiposol	postgres	
compara_vigencias	postgres	
compara_vigencias_tmp	postgres	
comparacion_efectividad_prerrequisitos	postgres	
concepto_ips	postgres	
conceptos_medicos	postgres	
contribuyentes	postgres	
cuestionario	postgres	
det_error_gest_prerrequisitos	postgres	
dictámenes	postgres	
documento_devolucion	postgres	
documento_pregunta	postgres	
documentos	postgres	
documentos_faltantes	postgres	
documentos_obligatorios	postgres	
documentos_pendientes_ips	postgres	

eliminamotivotmp	postgres	
entidades	postgres	
entidades_dictamenes	postgres	
entidades_gestion	postgres	
entidades_prerradicacion	postgres	
entidades_prerradicacion_temp	postgres	
entidades_prerradicacion_tmp	postgres	
entidades_tramite	postgres	
envios_documentales	postgres	
eps	postgres	
errores_comparacion	postgres	
errores_generan_tarea	postgres	
estados_analisis	postgres	
estados_bonos	postgres	
estados_imagine	postgres	
estructura_cierres_masivos	postgres	
fechas_vencimiento	postgres	
festivos	postgres	
gestion_analisis	postgres	
gestion_bandejas	postgres	
gestion_bonos	postgres	
gestion_codificaciones	postgres	
gestion_conceptos	postgres	
gestion_dictamenes_otrasent	postgres	
gestion_funeraria	postgres	
gestion_imagine	postgres	
gestion_it	postgres	
gestion_prerradicacion	postgres	
gestion_prerrequisitos	postgres	
gestion_solicitudes_ips	postgres	
historico_asignacion	postgres	
historico_reasignacion	postgres	
homologacion_estados	postgres	
imagenes_asociadas	postgres	
imagenes_prerradicacion	postgres	
indicador_errores	postgres	
log_cambio_fecharad	postgres	

log_edicion	postgres	
log_eliminacion	postgres	
log_pasos_analisis	postgres	
matching_prerrequisitos	postgres	
matching_prerrequisitos_tmp	postgres	
mensajes_obp	postgres	
mensajes_obp_tramite	postgres	
menu	postgres	
motivos_prerradicacion	postgres	
notificaciones	postgres	
oferta_servicio	postgres	
ofertas_bandejas	postgres	
oficinas	postgres	
panel_asignacion	postgres	
parentesco	postgres	
permisos_menu	postgres	
preguntas	postgres	
prerradicacion	postgres	
prerrequisitos_automaticos	postgres	
producto	postgres	
punteo_solicitudes	postgres	
reagendamiento	postgres	
recepcion_documentos	postgres	
recepcion_documentos_ips	postgres	
resp_cuestionario	postgres	
responsables	postgres	
responsables_bonos	postgres	
responsables_imagine	postgres	
responsables_tramite	postgres	
secciones_guia	postgres	
segundos_grupos	postgres	
solicitudes_rechazadas	postgres	
solicitudes_rechazadas_tmp	postgres	
subtipos	postgres	
tareas_inconsistencias	postgres	
tarifas	postgres	
tipos_beneficiario	postgres	

tipos_cobro	postgres	
tipos_documento	postgres	
tipos_prestacion	postgres	
tmp_rechazos	postgres	
trazabilidad_conceptos	postgres	
trazabilidad_prestaciones	postgres	
trazabilidad_prestaciones_elim	postgres	
usuarios	postgres	
validacion_punto_revision	postgres	

5.3.3.2. Diccionario de Datos

En la siguiente tabla se encuentra el listado de campos por tabla y el tipo de dato por campo y su tamaño.

Tabla 2 Diccionario de datos

Tabla	Columna	Restricciones	Tipo de Dato	Tamaño
acciones_legales	id	nextval('acciones_legales_id_seq'::regclass)	integer	
acciones_legales	nombre_accion		character varying	
acciones_legales	cierra_tareas		character varying	
acuerdos_servicio	na	nextval('acuerdos_servicio_na_seq'::regclass)	integer	
acuerdos_servicio	tipo_solicitud		character varying	
acuerdos_servicio	mesa_control1		integer	
acuerdos_servicio	precategorizacion		integer	
acuerdos_servicio	mesa_control2		integer	
acuerdos_servicio	categorizacion		integer	
acuerdos_servicio	validaciones_sitio		integer	

acuerdos_servicio	liquidaciones		integer	
acuerdos_servicio	historias_laborales		integer	
acuerdos_servicio	llamada_funeraria		integer	
acuerdos_servicio	crear_ben_grupos		integer	
acuerdos_servicio	precategorizacion_grupos		integer	
acuerdos_servicio	categorizacion_grupos		integer	
acuerdos_servicio	simulacion		integer	
adj_conceptos	id_adjunto	nextval('adj_conceptos_id_adjunto_seq'::regclass)	integer	
adj_conceptos	id_concepto		integer	
adj_conceptos	path_adjunto		character varying	
adj_dictamenes	id	nextval('adj_dictamenes_id_seq'::regclass)	integer	
adj_dictamenes	id_concepto		integer	
adj_dictamenes	path		character varying	
asignacion_bonos	id	nextval('asignacion_bonos_id_seq'::regclass)	integer	
asignacion_bonos	usuario		character varying	
asignacion_bonos	tramite		character varying	
asignacion_bonos	fecha_asignacion	now()	timestamp without time zone	
asignacion_gestion	id	nextval('asignacion_gestion_id_seq'::regclass)	integer	

asignacion_gestion	usuario		character varying	
asignacion_gestion	tramite		character varying	
asignacion_gestion	fecha_asignacion	now()	timestamp without time zone	
asignacion_imagine	id	nextval('asignacion_imagine_id_seq'::regclass)	integer	
asignacion_imagine	tramite		character varying	
asignacion_imagine	usuario		character varying	
asignacion_imagine	bandeja		character varying	
asignacion_imagine	fecha_asignacion	now()	timestamp without time zone	
asignacion_tramites	na	nextval('asignacion_tramites_na_seq'::regclass)	integer	
asignacion_tramites	tramite		character varying	
asignacion_tramites	usuario		character varying	
asignacion_tramites	proceso		character varying	
asignacion_tramites	fecha_asignacion	now()	timestamp without time zone	

asignacion_t ramites	tipo_solicitud		character varying	
asignacion_t ramites	categoria		character varying	
beneficiarios _prerradicacion	id_ben	nextval('beneficiarios_prerradicacion_id_ben_seq'::regclass)	integer	
beneficiarios _prerradicacion	cod_prerradicacion		character varying	
beneficiarios _prerradicacion	tipo_benef		integer	
beneficiarios _prerradicacion	tipo_sol		character varying	
beneficiarios _prerradicacion	identificacion_benef		character varying	
beneficiarios _prerradicacion	primer_nombre		character varying	
beneficiarios _prerradicacion	segundo_nombre		character varying	
beneficiarios _prerradicacion	primer_apellido		character varying	
beneficiarios _prerradicacion	segundo_apellido		character varying	
beneficiarios _prerradicacion	razon_social		character varying	
beneficiarios _prerradicacion	telefono		character varying	
beneficiarios _prerradicacion	celular		character varying	

beneficiarios _prerradicacion	autosms		character varying	
beneficiarios _prerradicacion	correo		character varying	
beneficiarios _prerradicacion	autoenvio		character varying	
beneficiarios _prerradicacion	ciudad		character varying	
beneficiarios _prerradicacion	tipodoc_beneficiario		character varying	
beneficiarios _prerradicacion	direccion_correspondencia		character varying	
beneficiarios _prerradicacion	fechanac_benef		character varying	
beneficiarios _prerradicacion	genero_ben		character varying	
beneficiarios _prerradicacion	calidad_benef		character varying	
beneficiarios _prerradicacion	parentesco		character varying	
beneficiarios _prerradicacion	user_insert		character varying	
beneficiarios _prerradicacion	fecha_insert	now()	timestamp without time zone	
beneficiarios _prerradicacion	estado_cruce		character varying	
beneficiarios _prerradicacion_temp	id_ben	nextval('beneficiarios_prerradicacion_temp_id_ben_seq'::regclass)	integer	

beneficiarios _prerradicacion_temp	cod_prerradicacion		character varying	
beneficiarios _prerradicacion_temp	tipo_benef		integer	
beneficiarios _prerradicacion_temp	tipo_sol		character varying	
beneficiarios _prerradicacion_temp	identificacion_benef		character varying	
beneficiarios _prerradicacion_temp	primer_nombre		character varying	
beneficiarios _prerradicacion_temp	segundo_nombre		character varying	
beneficiarios _prerradicacion_temp	primer_apellido		character varying	
beneficiarios _prerradicacion_temp	segundo_apellido		character varying	
beneficiarios _prerradicacion_temp	razon_social		character varying	
beneficiarios _prerradicacion_temp	telefono		character varying	
beneficiarios _prerradicacion_temp	celular		character varying	
beneficiarios _prerradicacion_temp	autosms		character varying	
beneficiarios _prerradicacion_temp	correo		character varying	
beneficiarios _prerradicacion_temp	autoenvio		character varying	

beneficiarios _prerradicacion_temp	ciudad		character varying	
beneficiarios _prerradicacion_temp	tipodoc_beneficiario		character varying	
beneficiarios _prerradicacion_temp	direccion_correspondencia		character varying	
beneficiarios _prerradicacion_temp	fechanac_benef		character varying	
beneficiarios _prerradicacion_temp	genero_ben		character varying	
beneficiarios _prerradicacion_temp	calidad_benef		character varying	
beneficiarios _prerradicacion_temp	parentesco		character varying	
beneficiarios _prerradicacion_temp	user_insert		character varying	
beneficiarios _prerradicacion_temp	fecha_insert	now()	timestamp without time zone	
beneficiarios _temp	id_temp		character varying	
beneficiarios _temp	tipo_benef		integer	
beneficiarios _temp	tiposol		character varying	
beneficiarios _temp	subtipo		character varying	
beneficiarios _temp	identificacion_benef		character varying	

beneficiarios_temp	primer_nombre		character varying	
beneficiarios_temp	segundo_nombre		character varying	
beneficiarios_temp	primer_apellido		character varying	
beneficiarios_temp	segundo_apellido		character varying	
beneficiarios_temp	razon_social		character varying	
beneficiarios_temp	telefono		character varying	
beneficiarios_temp	celular		character varying	
beneficiarios_temp	autosms		character varying	
beneficiarios_temp	correo		character varying	
beneficiarios_temp	autoenvio		character varying	
beneficiarios_temp	ciudad		character varying	
beneficiarios_temp	tipodoc_beneficiario		character varying	
beneficiarios_temp	direccion_correspondencia		character varying	
beneficiarios_temp	consecutivo	nextval('beneficiarios_temp_consecutivo_seq'::regclass)	integer	
casos_devueltos	id	nextval('casos_devueltos_id_seq'::regclass)	integer	

casos_devueltos	tramite		character varying	
casos_devueltos	bandeja		character varying	
causales_rechazo	id_causal	nextval('causales_rechazo_id_causal_seq'::regclass)	integer	
causales_rechazo	descripcion		character varying	
causales_rechazo	estado_cau	0	integer	
causales_rechazo	tipo_solicitud		character varying	
cerradas_por_cierre	id	nextval('cerradas_por_cierre_id_seq'::regclass)	integer	
cerradas_por_cierre	id_traza		integer	
cerradas_por_cierre	motivo_cierre		character varying	
cerradas_por_cierre	observacion		text	
cerradas_por_cierre	fecha_cierre	now()	timestamp without time zone	
cerradas_por_cierre	estado	0	integer	
ciudades	codigo		character varying	
ciudades	codigo_depto		character varying	
ciudades	nombre_depto		character varying	
ciudades	codigo_ciudad		character varying	

ciudades	nombre_ciudad		character varying	
clasificacion_tiposol	id_clas	nextval('clasificacion_tiposol_id_clas_seq'::regclass)	integer	
clasificacion_tiposol	codigo		character varying	
clasificacion_tiposol	tipo_solicitud		character varying	
clasificacion_tiposol	subtipo		integer	
clasificacion_tiposol	letra_prerradicacion		character varying	
clasificacion_tiposol	titulo_pdf		character varying	
clasificacion_tiposol	encabezado		text	
clasificacion_tiposol	notas		text	
compara_vigencias	na	nextval('compara_vigencias_na_seq'::regclass)	integer	
compara_vigencias	tramite		character varying	
compara_vigencias	fecha_efectividad_as		character varying	
compara_vigencias	fecha_efectividad_siafp		character varying	
compara_vigencias	fecha_efectividad_obp		character varying	
compara_vigencias	tipo_vinculacion_as		character varying	
compara_vigencias	tipo_vinculacion_siafp		character varying	

compara_vigencias	tipo_vinculacion_obp		character varying	
compara_vigencias	usuario_gestion		character varying	
compara_vigencias	fecha_insert	now()	timestamp without time zone	
compara_vigencias_tmp	na	nextval('compara_vigencias_tmp_na_seq'::regclass)	integer	
compara_vigencias_tmp	tramite		character varying	
compara_vigencias_tmp	fecha_efectividad_as		character varying	
compara_vigencias_tmp	fecha_efectividad_siafp		character varying	
compara_vigencias_tmp	fecha_efectividad_obp		character varying	
compara_vigencias_tmp	tipo_vinculacion_as		character varying	
compara_vigencias_tmp	tipo_vinculacion_siafp		character varying	
compara_vigencias_tmp	tipo_vinculacion_obp		character varying	
compara_vigencias_tmp	usuario_gestion		character varying	
comparacion_efectividad_prerrequisitos	na	nextval('comparacion_efectividad_prerrequisitos_na_seq'::regclass)	integer	
comparacion_efectividad_	tramite		character varying	

prerrequisito s				
comparacion _efectividad_ prerrequisito s	plataforma		character varying	
comparacion _efectividad_ prerrequisito s	fecha_efectividad		character varying	
comparacion _efectividad_ prerrequisito s	tipo_vinculacion		character varying	
comparacion _efectividad_ prerrequisito s	fecha_comparacion	now()	timestamp without time zone	
comparacion _efectividad_ prerrequisito s	usuario_comparacion		character varying	
concepto_ips	id	nextval('concepto_ips_id_seq'::regclass)	integer	
concepto_ips	pronositco		character varying	
concepto_ips	usuario		character varying	
concepto_ips	tramite		character varying	
concepto_ips	fecha_insert	now()	timestamp without time zone	
concepto_ips	observaciones		text	
concepto_ips	cruce_ips	'PENDIENTE CRUZAR'::character varying	character varying	

concepto_ips	fecha_cruce		timestamp without time zone	
concepto_ips	dia_remision		character varying	
conceptos_medicos	id_concepto	nextval('conceptos_medicos_id_concepto_seq'::regclass)	integer	
conceptos_medicos	tipo_documento		character varying	
conceptos_medicos	id_afiliado		character varying	
conceptos_medicos	primer_nombre		character varying	
conceptos_medicos	segundo_nombre		character varying	
conceptos_medicos	primer_apellido		character varying	
conceptos_medicos	segundo_apellido		character varying	
conceptos_medicos	eps		character varying	
conceptos_medicos	pronostico		character varying	
conceptos_medicos	incapacidad_anexa		character varying	
conceptos_medicos	fecha_remision		character varying	
conceptos_medicos	fecha_recibido		character varying	

conceptos_medicos	historia_anexa		character varying	
conceptos_medicos	digitalizado		character varying	
conceptos_medicos	direccion_eps		character varying	
conceptos_medicos	telefono_eps		character varying	
conceptos_medicos	ciudad		character varying	
conceptos_medicos	contacto_eps		character varying	
conceptos_medicos	especialidad		character varying	
conceptos_medicos	direccion_residencia		character varying	
conceptos_medicos	ciudad_residencia		character varying	
conceptos_medicos	telefono_residencia		character varying	
conceptos_medicos	telefono_adicional		character varying	
conceptos_medicos	email		character varying	
conceptos_medicos	autoriza_mail		character varying	
conceptos_medicos	envio_mail		character varying	
conceptos_medicos	fecha_insert	now()	timestamp	

			without time zone	
conceptos_ medicos	usuario_insert		character varying	
conceptos_ medicos	estado_concepto		character varying	
conceptos_ medicos	cie10		character varying	
conceptos_ medicos	num_no_contactado		integer	
conceptos_ medicos	conducta_it		character varying	
conceptos_ medicos	cruce_ips	'PENDIENTE CRUZAR':character varying	character varying	
conceptos_ medicos	fecha_cruce		timestamp without time zone	
conceptos_ medicos	tramite_padre		character varying	
contribuyentes	id	nextval('contribuyentes_id_seq':regclass)	integer	
contribuyentes	nit		character varying	
contribuyentes	nombre		character varying	
contribuyentes	clasificacion		character varying	
cuestionario	id	nextval('cuestionario_id_seq':regclass)	integer	
cuestionario	clas		character varying	

cuestionario	pregunta		character varying	
det_error_gest_prerrequisitos	na	nextval('det_error_gest_prerrequisitos_na_seq'::regclass)	integer	
det_error_gest_prerrequisitos	tarea		character varying	
det_error_gest_prerrequisitos	campo_error		character varying	
det_error_gest_prerrequisitos	tramite		character varying	
det_error_gest_prerrequisitos	usuario_insert		character varying	
det_error_gest_prerrequisitos	fecha_insert	now()	timestamp without time zone	
dictamenes	id	nextval('dictamenes_id_seq'::regclass)	integer	
dictamenes	tipo_doc		character varying	
dictamenes	num_doc		character varying	
dictamenes	primer_nombre		character varying	
dictamenes	segundo_nombre		character varying	
dictamenes	primer_apellido		character varying	
dictamenes	segundo_apellido		character varying	

dictámenes	entidad_dictamen		character varying	
dictámenes	fecha_dictamen		character varying	
dictámenes	fecha_recepcion		character varying	
dictámenes	tipo		character varying	
dictámenes	solo_origen		character varying	
dictámenes	origen		character varying	
dictámenes	res_dic		character varying	
dictámenes	usuario_insert		character varying	
dictámenes	fecha_insert	now()	timestamp without time zone	
dictámenes	gestionado	'NO'::character varying	character varying	
documento_devolucion	tramite		character varying	
documento_devolucion	id_documento_faltante		integer	
documento_devolucion	id_traza		integer	
documento_devolucion	gestionado	0	integer	
documento_devolucion	observacion		text	

documento_pregunta	id	nextval('documento_pregunta_id_seq'::regclass)	integer	
documento_pregunta	pregunta		character varying	
documento_pregunta	documento		character varying	
documento_pregunta	clas		character varying	
documento_pregunta	seccion		integer	
documentos	id_documento	nextval('documentos_id_documento_seq1'::regclass)	integer	
documentos	cod_documento		character varying	
documentos	descripcion		character varying	
documentos	validaciones		text	
documentos	obligatorio		integer	
documentos	tipo_documento		character varying	
documentos	tipo_gestion		character varying	
documentos	casuistica		text	
documentos_faltantes	id_documento_faltante	nextval('documentos_faltantes_id_documento_faltante_seq'::regclass)	integer	
documentos_faltantes	nombre_documento		character varying	
documentos_faltantes	activo	1	integer	
documentos_obligatorios	na	nextval('documentos_obligatorios_na_seq'::regclass)	integer	
documentos_obligatorios	clas		character varying	

documentos_obligatorios	documento		character varying	
documentos_obligatorios	seccion		integer	
documentos_pendientes_ips	na_documento	nextval('documentos_pendientes_ips_na_documento_seq'::regclass)	integer	
documentos_pendientes_ips	tramite		character varying	
documentos_pendientes_ips	documento		character varying	
documentos_pendientes_ips	fecha_insert	now()	timestamp without time zone	
documentos_pendientes_ips	user_insert		character varying	
documentos_pendientes_ips	estado_documento		character varying	
documentos_pendientes_ips	reporte	'Pendientes Enviar'::character varying	character varying	
eliminamotivotmp	consecutivo	nextval('eliminamotivotmp_consecutivo_seq'::regclass)	integer	
eliminamotivotmp	id_mot		integer	
eliminamotivotmp	user_insert		character varying	
eliminamotivotmp	tramite		character varying	
entidades	id	nextval('entidades_id_seq'::regclass)	integer	
entidades	nombre		character varying	

entidades	nit		character varying	
entidades	categoría		character varying	
entidades_dietamenes	cod_entidad		character varying	
entidades_dietamenes	nombre_entidad		character varying	
entidades_dietamenes	estado	0	integer	
entidades_gestion	na	nextval('entidades_gestion_na_seq'::regclass)	integer	
entidades_gestion	tramite		character varying	
entidades_gestion	nit		character varying	
entidades_gestion	razon_social		character varying	
entidades_gestion	estado		integer	
entidades_gestion	observacion		text	
entidades_gestion	user_insert		character varying	
entidades_gestion	fecha_insert	now()	timestamp without time zone	
entidades_prerradicacion	id	nextval('entidades_prerradicacion_id_seq'::regclass)	integer	
entidades_prerradicacion	cod_prerradicado		character varying	

entidades_pr erradicacion	tipo_historia		character varying	
entidades_pr erradicacion	motivo		integer	
entidades_pr erradicacion	entidad		character varying	
entidades_pr erradicacion	user_insert		character varying	
entidades_pr erradicacion	fecha_insert	now()	timestamp without time zone	
entidades_pr erradicacion	fecha_desde		character varying	
entidades_pr erradicacion	fecha_hasta		character varying	
entidades_pr erradicacion	salario_real		character varying	
entidades_pr erradicacion	cotiza_iss		character varying	
entidades_pr erradicacion	cotiza_otros		character varying	
entidades_pr erradicacion _temp	id	nextval('entidades_prerradicacion_tmp_id_seq'::regclass)	integer	
entidades_pr erradicacion _temp	cod_prerradica do		character varying	
entidades_pr erradicacion _temp	tipo_historia		character varying	
entidades_pr erradicacion _temp	motivo		integer	

entidades_pr erradicacion _tmp	entidad		character varying	
entidades_pr erradicacion _tmp	user_insert		character varying	
entidades_pr erradicacion _tmp	fecha_insert	now()	timestamp without time zone	
entidades_pr erradicacion _tmp	fecha_desde		character varying	
entidades_pr erradicacion _tmp	fecha_hasta		character varying	
entidades_pr erradicacion _tmp	salario_real		character varying	
entidades_pr erradicacion _tmp	cotiza_iss		character varying	
entidades_pr erradicacion _tmp	cotiza_otros		character varying	
entidades_pr erradicacion _tmp	id	nextval('entidades_prerradicacion_tmp_id_seq1'::regclass)	integer	
entidades_pr erradicacion _tmp	guid		character varying	
entidades_pr erradicacion _tmp	tipo_historia		character varying	
entidades_pr erradicacion _tmp	motivo		integer	
entidades_pr erradicacion _tmp	entidad		character varying	
entidades_pr erradicacion _tmp	user_insert		character varying	

entidades_pr erradicacion _tmp	fecha_desde		character varying	
entidades_pr erradicacion _tmp	fecha_hasta		character varying	
entidades_pr erradicacion _tmp	salario_real		character varying	
entidades_pr erradicacion _tmp	cotiza_iss		character varying	
entidades_pr erradicacion _tmp	cotiza_otros		character varying	
entidades_tr amite	na	nextval('entidades_tramite_na_ seq'::regclass)	integer	
entidades_tr amite	tramite		character varying	
entidades_tr amite	nit		character varying	
entidades_tr amite	razon_social		character varying	
entidades_tr amite	estado		integer	
entidades_tr amite	observacion		text	
entidades_tr amite	user_insert		character varying	
entidades_tr amite	fecha_insert	now()	timestamp without time zone	
envios_docu mentales	id_envio	nextval('envios_documentales_i d_envio_seq'::regclass)	integer	
envios_docu mentales	cod_prerradica do		character varying	

envios_documentales	num_folios		integer	
envios_documentales	oficina		character varying	
envios_documentales	usuario		character varying	
envios_documentales	fecha_envio	now()	timestamp without time zone	
envios_documentales	estado		character varying	
envios_documentales	usuario_imagin	e	character varying	
envios_documentales	fecha_recepcion		timestamp without time zone	
envios_documentales	folios_recibidos		integer	
eps	codigo		character varying	
eps	nit		character varying	
eps	nombre		character varying	
eps	estado		integer	
errores_comparacion	na	nextval('errores_comparacion_na_seq'::regclass)	integer	
errores_comparacion	tramite		character varying	
errores_comparacion	error		character varying	

errores_comparacion	fecha_insert	now()	timestamp without time zone	
errores_comparacion	usuario_insert		character varying	
errores_generacion_tarea	error		character varying	
errores_generacion_tarea	area_responsable		character varying	
estados_analisis	na	nextval('estados_analisis_na_seq'::regclass)	integer	
estados_analisis	descripcion		character varying	
estados_analisis	bandeja		character varying	
estados_analisis	cierra_tarea		character varying	
estados_analisis	cambia_bandeja		character varying	
estados_bonos	na	nextval('estados_bonos_na_seq'::regclass)	integer	
estados_bonos	descripcion		character varying	
estados_bonos	bandeja		character varying	
estados_bonos	origen		character varying	
estados_bonos	cierra_tarea		character varying	

estados_bonos	cambia_bandeja		character varying	
estados_imagine	na	nextval('estados_imagine_na_seq'::regclass)	integer	
estados_imagine	descripcion		character varying	
estados_imagine	bandeja		character varying	
estados_imagine	cierra_tarea		character varying	
estados_imagine	cambia_bandeja		character varying	
estructura_cierres_masivos	id	nextval('estructura_cierres_masivos_id_seq'::regclass)	integer	
estructura_cierres_masivos	bandeja		character varying	
estructura_cierres_masivos	nombre_campo		character varying	
estructura_cierres_masivos	lista_valor		character varying	
estructura_cierres_masivos	descripcion_valor		character varying	
estructura_cierres_masivos	orden_campo		integer	
fechas_vencimiento	id	nextval('fechas_vencimiento_id_seq'::regclass)	integer	
fechas_vencimiento	tipo_solicitud		character varying	
fechas_vencimiento	dias		integer	

fechas_vencimiento	tipo_plazo		character varying	
festivos	dia		character varying	
festivos	tipo		character	1
gestion_analisis	id_gestion	nextval('gestion_analisis_id_gestion_seq'::regclass)	integer	
gestion_analisis	tramite		character varying	
gestion_analisis	estado		integer	
gestion_analisis	observacion		text	
gestion_analisis	fecha_insert	now()	timestamp without time zone	
gestion_analisis	usuario_insert		character varying	
gestion_analisis	accion_legal		character varying	
gestion_analisis	pago_ant_bono		character varying	
gestion_analisis	pago_ant_cobro		character varying	
gestion_analisis	notificado_por		character varying	
gestion_bandejas	id_gestion	nextval('gestion_bandejas_id_gestion_seq'::regclass)	integer	
gestion_bandejas	tramite		character varying	
gestion_bandejas	estado_gestion		integer	

gestion_ban dejas	observacion		text	
gestion_ban dejas	user_gestion		character varying	
gestion_ban dejas	fecha_gestion	now()	timestamp without time zone	
gestion_ban dejas	origen_gestion		character varying	
gestion_ban dejas	sub_estado		character varying	
gestion_ban dejas	entidad		integer	
gestion_bon os	id_gestion	nextval('gestion_bonos_id_gestion_seq'::regclass)	integer	
gestion_bon os	tramite		character varying	
gestion_bon os	estado_gestion		integer	
gestion_bon os	observacion		text	
gestion_bon os	aplica_ent		character varying	
gestion_bon os	id_entidad		integer	
gestion_bon os	user_gestion		character varying	
gestion_bon os	fecha_gestion	now()	timestamp without time zone	
gestion_bon os	origen_gestion		character varying	

gestion_codificaciones	consecutivo	nextval('gestion_codificaciones_consecutivo_seq'::regclass)	integer	
gestion_codificaciones	codificacion_actual		character varying	
gestion_codificaciones	codificacion_futura		character varying	
gestion_codificaciones	bandeja		character varying	
gestion_conceptos	id_gestion	nextval('gestion_conceptos_id_gestion_seq'::regclass)	integer	
gestion_conceptos	contacta		character varying	
gestion_conceptos	fecha_problabre_rad		character varying	
gestion_conceptos	fallecido		character varying	
gestion_conceptos	observacion		text	
gestion_conceptos	usuario_gestion		character varying	
gestion_conceptos	fecha_gestion	now()	timestamp without time zone	
gestion_conceptos	id_concepto		integer	
gestion_conceptos	area_gestion	'Call Center'::character varying	character varying	
gestion_conceptos	origen_gestion		character varying	
gestion_conceptos	cod_futura		integer	

gestion_dicta menes_otras ent	id	nextval('gestion_dictamenes_ot rasent_id_seq'::regclass)	integer	
gestion_dicta menes_otras ent	id_dictamen		integer	
gestion_dicta menes_otras ent	fecha_insert	now()	timesta mp without time zone	
gestion_dicta menes_otras ent	usuario		charact er varying	
gestion_dicta menes_otras ent	observacion		charact er varying	
gestion_dicta menes_otras ent	estado		charact er varying	
gestion_fune raria	id_gestion	nextval('gestion_funeraria_id_g estion_seq'::regclass)	integer	
gestion_fune raria	id_reclamante		integer	
gestion_fune raria	tramite		charact er varying	
gestion_fune raria	estado		charact er varying	
gestion_fune raria	observacion		text	
gestion_fune raria	contacto_gestio n		charact er varying	
gestion_fune raria	reclama_paga		charact er varying	
gestion_fune raria	poder		charact er varying	
gestion_fune raria	pagado_por		charact er varying	

gestion_fune raria	usuario_gestion		character varying	
gestion_fune raria	fecha_gestion	now()	timestamp without time zone	
gestion_fune raria	plan_exequial		character varying	
gestion_fune raria	numero_plan		character varying	
gestion_fune raria	titular_fallecido		character varying	
gestion_fune raria	nombre_funera ria		character varying	
gestion_fune raria	estado_cruce		character varying	
gestion_ima gine	id_gestion	nextval('gestion_imagine_id_ge stion_seq'::regclass)	integer	
gestion_ima gine	tramite		character varying	
gestion_ima gine	estado		integer	
gestion_ima gine	observacion		text	
gestion_ima gine	fecha_insert	now()	timestamp without time zone	
gestion_ima gine	usuario_insert		character varying	
gestion_it	id_gestion	nextval('gestion_it_id_gestion_s eq'::regclass)	integer	

gestion_it	tramite		character varying	
gestion_it	estado		character varying	
gestion_it	observacion		text	
gestion_it	fecha_insert	now()	timestamp without time zone	
gestion_it	usuario_insert		character varying	
gestion_it	ipp		integer	
gestion_prerradicacion	id_gestion	nextval('gestion_prerradicacion_id_gestion_seq'::regclass)	integer	
gestion_prerradicacion	tramite		character varying	
gestion_prerradicacion	contacta		character varying	
gestion_prerradicacion	observacion		text	
gestion_prerradicacion	usuario_gestion		character varying	
gestion_prerradicacion	fecha_gestion	now()	timestamp without time zone	
gestion_prerradicacion	tipo_gestion		character varying	
gestion_prerradicacion	cod_futura		integer	
gestion_prerradicacion	fecha_agenda		character varying	
gestion_prerrequisitos	na_gestion	nextval('gestion_prerrequisitos_na_gestion_seq'::regclass)	integer	

gestion_prerrequisitos	tramite		character varying	
gestion_prerrequisitos	area		character varying	
gestion_prerrequisitos	tarea		character varying	
gestion_prerrequisitos	respuesta_tarea		character varying	
gestion_prerrequisitos	observacion		text	
gestion_prerrequisitos	fecha_insert	now()	timestamp without time zone	
gestion_prerrequisitos	usuario_insert		character varying	
gestion_prerrequisitos	genera_tarea		character varying	
gestion_prerrequisitos	tipopro		character varying	
gestion_prerrequisitos	emisor		character varying	
gestion_prerrequisitos	fecha_estructuracion		character varying	
gestion_prerrequisitos	porcentaje_pcl		character varying	
gestion_prerrequisitos	etapa	'Prerradicacion'::character varying	character varying	
gestion_solicitudes_ips	id_gestion	nextval('gestion_solicitudes_ips_id_gestion_seq'::regclass)	integer	

gestion_solicitudes_ips	tramite		character varying	
gestion_solicitudes_ips	devolucion		bigint	
gestion_solicitudes_ips	estado_gestion		character varying	
gestion_solicitudes_ips	contacto		character varying	
gestion_solicitudes_ips	telefono		double precision	
gestion_solicitudes_ips	observacion		text	
gestion_solicitudes_ips	fecha_insert		timestamp without time zone	
gestion_solicitudes_ips	usuario_insert		character varying	
gestion_solicitudes_ips	email		character varying	
gestion_solicitudes_ips	direccion		character varying	
gestion_solicitudes_ips	celular		double precision	
historico_asignacion	id	nextval('historico_asignacion_id_seq'::regclass)	integer	
historico_asignacion	tramite		character varying	
historico_asignacion	usuario		character varying	
historico_asignacion	fecha_asignacion	now()	timestamp without	

			time zone	
historico_asignacion	bandeja		character varying	
historico_reasignacion	id	nextval('historico_reasignacion_id_seq'::regclass)	integer	
historico_reasignacion	tramite		character varying	
historico_reasignacion	usuario_asignado		character varying	
historico_reasignacion	usuario_asigna		character varying	
historico_reasignacion	fecha_asignacion	now()	timestamp without time zone	
homologacion_estados	id	nextval('homologacion_estados_id_seq'::regclass)	integer	
homologacion_estados	estado_principal		character varying	
homologacion_estados	sub_estado		character varying	
homologacion_estados	sub_estado_traza		character varying	
homologacion_estados	estado_tarea		character varying	
imagenes_asociadas	id	nextval('imagenes_asociadas_id_seq'::regclass)	integer	
imagenes_asociadas	id_imagen		integer	
imagenes_asociadas	tramite_nuevo		character varying	
imagenes_asociadas	fecha_asociacion	now()	timestamp	

			without time zone	
imagenes_a sociadas	usuario_asocia		charact er varying	
imagenes_pr erradicacion	id	nextval('imagenes_prerradicaci on_id_seq'::regclass)	integer	
imagenes_pr erradicacion	cod_prerradica do		charact er varying	
imagenes_pr erradicacion	path		charact er varying	
imagenes_pr erradicacion	user_insert		charact er varying	
imagenes_pr erradicacion	fecha_insert	now()	timesta mp without time zone	
imagenes_pr erradicacion	origen_imagen		charact er varying	
indicador_err ores	na	nextval('indicador_errores_na_ seq'::regclass)	integer	
indicador_err ores	bandeja_origen		charact er varying	
indicador_err ores	bandeja_destin o		charact er varying	
indicador_err ores	tramite		charact er varying	
indicador_err ores	rechazo		charact er varying	
indicador_err ores	obs_rechazo		text	
indicador_err ores	fecha_rechazo	now()	timesta mp without	

			time zone	
indicador_errores	usuario_rechaza		character varying	
log_cambio_fecha rad	sr	nextval('log_cambio_fecha rad_sr_seq'::regclass)	integer	
log_cambio_fecha rad	fecha_inser	now()	time stamp without time zone	
log_cambio_fecha rad	tramite		character varying	
log_cambio_fecha rad	fecha_ant		character varying	
log_cambio_fecha rad	fecha_act		character varying	
log_cambio_fecha rad	usuario_mod		character varying	
log_edicion	id_edicion	nextval('log_edicion_id_edicion _seq'::regclass)	integer	
log_edicion	tramite		character varying	
log_edicion	user_edita		character varying	
log_edicion	fecha_edita	now()	time stamp without time zone	
log_eliminacion	consecutivo	nextval('log_eliminacion_conse cutivo_seq'::regclass)	integer	
log_eliminacion	tramite		character varying	

log_eliminacion	motivo		character varying	
log_eliminacion	usuario_insert		character varying	
log_eliminacion	fecha_insert	now()	timestamp without time zone	
log_pasos_analisis	na	nextval('log_pasos_analisis_na_seq'::regclass)	integer	
log_pasos_analisis	tramite		character varying	
log_pasos_analisis	usuario		character varying	
log_pasos_analisis	fecha_modificacion	now()	timestamp without time zone	
matching_prerrequisitos	na	nextval('matching_prerrequisitos_na_seq'::regclass)	integer	
matching_prerrequisitos	tramite		character varying	
matching_prerrequisitos	primer_nombre		character varying	
matching_prerrequisitos	segundo_nombre		character varying	
matching_prerrequisitos	primer_apellido		character varying	
matching_prerrequisitos	segundo_apellido		character varying	
matching_prerrequisitos	fecha_nacimiento		character varying	

matching_prerrequisitos	usuario_inserta		character varying	
matching_prerrequisitos	tipo_proceso		character varying	
matching_prerrequisitos	genero		character varying	
matching_prerrequisitos	fecha_gestion	now()	timestamp without time zone	
matching_prerrequisitos_tmp	na	nextval('matching_prerrequisitos_tmp_na_seq'::regclass)	integer	
matching_prerrequisitos_tmp	tramite		character varying	
matching_prerrequisitos_tmp	primer_nombre		character varying	
matching_prerrequisitos_tmp	segundo_nombre		character varying	
matching_prerrequisitos_tmp	primer_apellido		character varying	
matching_prerrequisitos_tmp	segundo_apellido		character varying	
matching_prerrequisitos_tmp	fecha_nacimiento		character varying	
matching_prerrequisitos_tmp	usuario_inserta		character varying	
matching_prerrequisitos_tmp	tipo_proceso		character varying	
matching_prerrequisitos_tmp	genero		character varying	

mensajes_obp	id	nextval('mensajes_obp_id_seq':regclass)	integer	
mensajes_obp	mensaje		text	
mensajes_obp	estado	1	integer	
mensajes_obp	cod_mensaje		character varying	
mensajes_obp_tramite	na	nextval('mensajes_obp_tramite_na_seq':regclass)	integer	
mensajes_obp_tramite	tramite		character varying	
mensajes_obp_tramite	id_mensaje		integer	
mensajes_obp_tramite	fecha_insert	now()	timestamp without time zone	
mensajes_obp_tramite	usuario_insert		character varying	
menu	id	nextval('menu_id_seq':regclass)	integer	
menu	status		character varying	
menu	padre		integer	
menu	nombre		character varying	
menu	icono		character varying	
menu	url		character varying	
menu	orden		integer	
menu	padre_ppal		integer	
motivos_preradicacion	id	nextval('motivos_preradicacion_id_seq':regclass)	integer	

motivos_preradicacion	descripcion		character varying	
notificaciones	na	nextval('notificaciones_na_seq':regclass)	integer	
notificaciones	tramite		character varying	
notificaciones	tipo_notificacion		character varying	
notificaciones	fecha_envio	now()	timestamp without time zone	
notificaciones	usuario_envio		character varying	
notificaciones	mail_envio		character varying	
notificaciones	path		character varying	
oferta_servicio	na	nextval('oferta_servicio_na_seq':regclass)	integer	
oferta_servicio	tipo_solicitud		character varying	
oferta_servicio	oferta_mesa		integer	
oferta_servicio	oferta_prerrequisitos		integer	
oferta_servicio	oferta_prerradicacion		integer	
oferta_servicio	oferta_precategorización		integer	
oferta_servicio	plazo_legal		integer	
ofertas_bandejas	id_oferta	nextval('ofertas_bandejas_id_oferta_seq':regclass)	integer	

ofertas_bandejas	bandeja		character varying	
ofertas_bandejas	tiposol		character varying	
ofertas_bandejas	tipo_oferta		character varying	
ofertas_bandejas	tiempo_dias		integer	
ofertas_bandejas	asignado_por		character varying	
ofertas_bandejas	fecha_asignacion_oferta	now()	timestamp without time zone	
oficinas	codigo		character varying	
oficinas	titulo		character varying	
oficinas	regional_cod		character varying	
oficinas	regional		character varying	
panel_asignacion	id	nextval('panel_asignacion_id_seq'::regclass)	integer	
panel_asignacion	usuario		character varying	
panel_asignacion	tiposol		character varying	
panel_asignacion	regional		character varying	

panel_asignacion	sub_estado		character varying	
panel_asignacion	origen		character varying	
panel_asignacion	estado		integer	
panel_asignacion	responsable		character varying	
parentesco	id_parentesco	nextval('parentesco_id_parentesco_seq'::regclass)	integer	
parentesco	parentesco		character varying	
permisos_menu	id	nextval('permisos_menu_id_seq'::regclass)	integer	
permisos_menu	usuario		character varying	
permisos_menu	menu		integer	
preguntas	cod_pregunta		character varying	
preguntas	pregunta		character varying	
preguntas	estado		integer	
preguntas	codigo_agrupacion		character varying	
preguntas	orden		integer	
prerradicacion	id		character varying	
prerradicacion	tipodoc		character varying	
prerradicacion	telefono		character varying	

prerradicacion	celular		bigint	
prerradicacion	autorizasms		character varying	
prerradicacion	autoriza_envio		character varying	
prerradicacion	ciudad		character varying	
prerradicacion	tiposol		character varying	
prerradicacion	his_lab_bono		character varying	
prerradicacion	mot_his_lab_bono		text	
prerradicacion	his_lab_prot		character varying	
prerradicacion	mot_his_lab_prot		text	
prerradicacion	his_lab_otras		character varying	
prerradicacion	mot_his_lab_otras		text	
prerradicacion	usuario_insert		character varying	
prerradicacion	fecha_insert	now()	timestamp without time zone	
prerradicacion	estado		character varying	
prerradicacion	correo		character varying	

prerradicacion	prima_media		character varying	
prerradicacion	enti_publica_desde		character varying	
prerradicacion	enti_publica_hasta		character varying	
prerradicacion	subtipo		integer	
prerradicacion	primer_nombre		character varying	
prerradicacion	segundo_nombre		character varying	
prerradicacion	primer_apellido		character varying	
prerradicacion	segundo_apellido		character varying	
prerradicacion	fecha_vencimiento		character varying	
prerradicacion	ent_histlabbono		integer	
prerradicacion	ent_histlabprot		integer	
prerradicacion	ent_histlabotras		integer	
prerradicacion	ind_normalizacion		character	1
prerradicacion	ent_normalizacion		character varying	
prerradicacion	derecho_bono		character	1
prerradicacion	tramite_bono		character	1
prerradicacion	entidad_contribuyente		character varying	

prerradicacion	investigacion		character	1
prerradicacion	categoria	'PREAPROBADO'::character varying	character varying	
prerradicacion	estado_documentos	'Cargar Documentos'::character varying	character varying	
prerradicacion	direccion_corresp		character varying	
prerradicacion	dias_incapacitado	0	integer	
prerradicacion	tipo_vinculacion		character varying	
prerradicacion	rec_apoderado		character varying	
prerradicacion	nom_apoderado		character varying	
prerradicacion	numero_grupo	1	integer	
prerradicacion	saldo		character varying	
prerradicacion	folofc		integer	
prerradicacion	foleps		integer	
prerradicacion	producto		character varying	
prerradicacion	oficina		character varying	
prerradicacion	estado_radicion	'PENDIENTE RADICAR'::character varying	character varying	
prerradicacion	user_radica		character varying	

prerradicacion	fecha_radica		timestamp without time zone	
prerradicacion	estado_general		character varying	
prerradicacion	documentos_oficina		character varying	
prerradicacion	documentos_ips		character varying	
prerradicacion	observacion_ips		text	
prerradicacion	categoria_real		character varying	
prerradicacion	responsable		character varying	
prerradicacion	fecha_asignacion		timestamp without time zone	
prerradicacion	vencimiento_legal		character varying	
prerradicacion	estado_analisis	1	integer	
prerradicacion	cedula		character varying	
prerradicacion	tipo_registro	'Dia a Dia':character varying	character varying	
prerradicacion	fecha_radicion_sistema		timestamp without time zone	

prerradicacion	carga_analisis		character varying	
prerradicacion	tipo_visita		character varying	
prerradicacion	estado_marca		character varying	
prerradicacion	tramite_reactiva		character varying	
prerradicacion	marca_tutela		character varying	
prerradicacion	observacion_tutela		text	
prerradicacion	fecha_marca_tutela		timestamp without time zone	
prerradicacion	usuario_marca_tutela		character varying	
prerradicacion	fecha_marca_estado		timestamp without time zone	
prerradicacion	observacion_estado		text	
prerradicacion	usuario_marca_estado		character varying	
prerradicacion	estado_grupo		character varying	
prerradicacion	marca_grupos	'N':character varying	character varying	
prerradicacion	bono_post_analisis		character varying	

prerradicacion	analisis_post_bono		character varying	
prerradicacion	caso_reabierto		character varying	
prerradicacion	fecha_reapertura		timestamp without time zone	
prerradicacion	observacion_reapertura		text	
prerradicacion	usuario_reapertura		character varying	
prerradicacion	analisis_doc_pend	'N'::character varying	character varying	
prerradicacion	ind_post_analisis_grupo	'N'::character varying	character varying	
prerradicacion	grupo_activo	'N'::character varying	character varying	
prerrequisitos_automaticos	na	nextval('prerrequisitos_automaticos_na_seq'::regclass)	integer	
prerrequisitos_automaticos	id_afiliado		character varying	
prerrequisitos_automaticos	tipo_solicitud		character varying	
prerrequisitos_automaticos	rezagos		character varying	
prerrequisitos_automaticos	obs_rezagos		character varying	
prerrequisitos_automaticos	planillas		character varying	

prerrequisitos_automaticos	observacion_planillas		character varying	
prerrequisitos_automaticos	depositos		character varying	
prerrequisitos_automaticos	observacion_depositos		character varying	
prerrequisitos_automaticos	deudas		character varying	
prerrequisitos_automaticos	observacion_deudas		character varying	
prerrequisitos_automaticos	estado	'Cargado':character varying	character varying	
producto	codigo		character varying	
producto	descripcion		character varying	
producto	estado	0	integer	
punteo_solicitudes	na	nextval('punteo_solicitudes_na_seq'::regclass)	integer	
punteo_solicitudes	tramite		character varying	
punteo_solicitudes	folios_recibidos		integer	
punteo_solicitudes	usuario_recibe		character varying	
punteo_solicitudes	fecha_punteo	now()	timestamp without time zone	
punteo_solicitudes	estado_punteo		character varying	

punteo_solicitudes	numero_grupo		integer	
punteo_solicitudes	sub_estado		character varying	
reagendamiento	id	nextval('reagendamiento_id_seq'::regclass)	integer	
reagendamiento	tramite		character varying	
reagendamiento	fecha_inicial_agenda		timestamp without time zone	
reagendamiento	oficina		character varying	
reagendamiento	asesor		character varying	
reagendamiento	fecha_solicitud_reagendar	now()	timestamp without time zone	
reagendamiento	usuario_solicitud_reagendar		character varying	
recepcion_documentos	na	nextval('recepcion_documentos_na_seq'::regclass)	integer	
recepcion_documentos	na_documento		integer	
recepcion_documentos	tipo_documento		character varying	
recepcion_documentos	usuario_recibe		character varying	
recepcion_documentos	fecha_recibe	now()	timestamp without time zone	

repcion_documento	usuario_recibe_imagine		character varying	
repcion_documento	fecha_recibe_imagine		timestamp without time zone	
repcion_documento	estado_recibe		character varying	
repcion_documento	estado_recibe_imagine		character varying	
repcion_documento	tramite		character varying	
repcion_documento	id_afiliado		character varying	
repcion_documento	tipo_solicitud		character varying	
repcion_documento_ips	id	nextval('repcion_documento_ips_id_seq'::regclass)	integer	
repcion_documento_ips	na_documento		integer	
repcion_documento_ips	usuario_recibe_oficina		character varying	
repcion_documento_ips	fecha_recibe_oficina	now()	timestamp without time zone	
repcion_documento_ips	usuario_recibe_ips		character varying	
repcion_documento_ips	fecha_recibe_ips		timestamp without	

			time zone	
recepcion_documento_ips	estado_recepcion		character varying	
recepcion_documento_ips	observacion_ips		text	
resp_cuestionario	cod_prerradicado		character varying	
resp_cuestionario	cod_pregunta		character varying	
resp_cuestionario	respuesta		character varying	
responsables	id_responsable	nextval('responsables_id_responsable_seq'::regclass)	integer	
responsables	usuario		character varying	
responsables	tipo_solicitud		character varying	
responsables	categoria		character varying	
responsables	bandeja		character varying	
responsables	activo	1	integer	
responsables_bonos	id_responsable	nextval('responsables_bonos_id_responsable_seq'::regclass)	integer	
responsables_bonos	usuario		character varying	
responsables_bonos	regional		character varying	
responsables_bonos	estado	1	integer	

responsable s_imagine	id_resp	nextval('responsables_imagine _id_resp_seq'::regclass)	integer	
responsable s_imagine	usuario		character varying	
responsable s_imagine	sede		character varying	
responsable s_imagine	bandeja		character varying	
responsable s_imagine	estado		integer	
responsable s_tramite	id_responsable	nextval('responsables_tramite_i d_responsable_seq'::regclass)	integer	
responsable s_tramite	usuario		character varying	
responsable s_tramite	categoria		character varying	
responsable s_tramite	estado	1	integer	
secciones_g uia	na	nextval('secciones_guia_na_se q'::regclass)	integer	
secciones_g uia	descripcion		character varying	
secciones_g uia	orden		integer	
segundos_gr upos	na	nextval('segundos_grupos_na_ seq'::regclass)	integer	
segundos_gr upos	tramite		character varying	
segundos_gr upos	estado_grupo		character varying	
segundos_gr upos	fecha_insert	now()	timestamp without time zone	

segundos_grupos	usuario_insert		character varying	
segundos_grupos	numero_grupo		integer	
segundos_grupos	creado_as		character varying	
solicitudes_rechazadas	consecutivo	nextval('solicitudes_rechazadas_consecutivo_seq'::regclass)	integer	
solicitudes_rechazadas	tramite		character varying	
solicitudes_rechazadas	documento		character varying	
solicitudes_rechazadas	causal		bigint	
solicitudes_rechazadas	ccben		character varying	
solicitudes_rechazadas	observacion		text	
solicitudes_rechazadas	usuario_rechazo		character varying	
solicitudes_rechazadas	fecha_rechazo	now()	timestamp without time zone	
solicitudes_rechazadas	estado	0	integer	
solicitudes_rechazadas_tmp	consecutivo	nextval('solicitudes_rechazadas_tmp_consecutivo_seq'::regclass)	integer	
solicitudes_rechazadas_tmp	tramite		character varying	
solicitudes_rechazadas_tmp	documento		character varying	

solicitudes_rechazadas_t mp	causal		bigint	
solicitudes_rechazadas_t mp	ccben		character varying	
solicitudes_rechazadas_t mp	observacion		text	
solicitudes_rechazadas_t mp	usuario_rechazo		character varying	
solicitudes_rechazadas_t mp	fecha_rechazo	now()	timestamp without time zone	
solicitudes_rechazadas_t mp	estado	0	integer	
subtipos	id	nextval('subtipos_id_seq'::regclass)	integer	
subtipos	codigo		character varying	
subtipos	tipo		character varying	
subtipos	descripcion		character varying	
tareas_inconsistencias	id_tarea	nextval('tareas_inconsistencias_id_tarea_seq'::regclass)	integer	
tareas_inconsistencias	nombre		character varying	
tareas_inconsistencias	proceso_relacionado		character varying	
tarifas	sr	nextval('tarifas_sr_seq'::regclass)	integer	
tarifas	tipo_solicitud		character varying	

tarifas	proceso		character varying	
tarifas	tarifa		integer	
tipos_beneficiario	id	nextval('tipos_beneficiario_id_seq'::regclass)	integer	
tipos_beneficiario	beneficiario		character varying	
tipos_cobro	id	nextval('tipos_cobro_id_seq'::regclass)	integer	
tipos_cobro	tramite		character varying	
tipos_cobro	tipo_cobro		character varying	
tipos_documento	tipo		character varying	
tipos_documento	nombre_tipo		character varying	
tipos_prestacion	codigo		character varying	3
tipos_prestacion	descripcion		character varying	20
tmp_rechazos	id	nextval('tmp_rechazos_id_seq'::regclass)	integer	
tmp_rechazos	tramite		character varying	
tmp_rechazos	documento		character varying	
tmp_rechazos	causal		bigint	
tmp_rechazos	ccben		character varying	
tmp_rechazos	observacion		text	

tmp_rechazos	usuario_rechazo		character varying	
tmp_rechazos	fecha_rechazo	now()	timestamp without time zone	
trazabilidad_conceptos	consecutivo	nextval('trazabilidad_conceptos_consecutivo_seq'::regclass)	integer	
trazabilidad_conceptos	concepto		character varying	
trazabilidad_conceptos	proceso		character varying	
trazabilidad_conceptos	estado_actual		character varying	
trazabilidad_conceptos	estado_precedente		character varying	
trazabilidad_conceptos	fecha_insert	now()	timestamp without time zone	
trazabilidad_conceptos	usuario_insert		character varying	
trazabilidad_conceptos	sub_estado		character varying	
trazabilidad_conceptos	estado_tarea	'Abierta'::character varying	character varying	
trazabilidad_conceptos	usuario_cierre		character varying	
trazabilidad_conceptos	fecha_cierre		timestamp without time zone	

trazabilidad_conceptos	fecha_vencimiento		character varying	
trazabilidad_conceptos	origen		character varying	
trazabilidad_prestaciones	consecutivo	nextval('trazabilidad_prestaciones_consecutivo_seq'::regclass)	integer	
trazabilidad_prestaciones	tramite		character varying	
trazabilidad_prestaciones	proceso		character varying	
trazabilidad_prestaciones	estado_actual		character varying	
trazabilidad_prestaciones	estado_precedente		character varying	
trazabilidad_prestaciones	fecha_insert	now()	timestamp without time zone	
trazabilidad_prestaciones	usuario_insert		character varying	
trazabilidad_prestaciones	sub_estado		character varying	
trazabilidad_prestaciones	estado_tarea	'Abierta'::character varying	character varying	
trazabilidad_prestaciones	usuario_cierre		character varying	
trazabilidad_prestaciones	fecha_cierre		timestamp without time zone	

trazabilidad_ prestaciones	fecha_vencimie nto		charact er varying	
trazabilidad_ prestaciones	origen		charact er varying	
trazabilidad_ prestaciones _elim	consecutivo		bigint	
trazabilidad_ prestaciones _elim	tramite		charact er varying	
trazabilidad_ prestaciones _elim	proceso		charact er varying	
trazabilidad_ prestaciones _elim	estado_actual		charact er varying	
trazabilidad_ prestaciones _elim	estado_preced ente		charact er varying	
trazabilidad_ prestaciones _elim	fecha_insert	now()	timesta mp without time zone	
trazabilidad_ prestaciones _elim	usuario_insert		charact er varying	
trazabilidad_ prestaciones _elim	sub_estado		charact er varying	
trazabilidad_ prestaciones _elim	estado_tarea	'Abierta':::character varying	charact er varying	
trazabilidad_ prestaciones _elim	usuario_cierre		charact er varying	
trazabilidad_ prestaciones _elim	fecha_cierre		timesta mp without time zone	

trazabilidad_prestaciones_elim	fecha_vencimiento		character varying	
trazabilidad_prestaciones_elim	origen		character varying	
usuarios	id	nextval('usuarios_id_seq'::regclass)	integer	
usuarios	usuario		character varying	
usuarios	nombre		character varying	
usuarios	area		character varying	
usuarios	estado		integer	
usuarios	sede		character varying	
validacion_punto_revision	id	nextval('validacion_punto_revision_id_seq'::regclass)	integer	
validacion_punto_revision	id_trazabilidad_prestaciones		integer	
validacion_punto_revision	estado		integer	

5.4. Descripción del Sistema

El sistema de control de pensionados tiene por objeto llevar un control de todas y cada una de las solicitudes de pensión radicadas para el cliente Protección Pensiones y Cesantías, el proyecto consiste en la implementación de un nuevo modelo de servicio y de software que involucra todas las áreas de la compañía, el software trabaja según el esquema de un workflow, que va desde la asesoría preliminar de la solicitud, hasta la notificación del pago de la misma o su respectivo rechazo.

El flujo permite llevar un control en línea de las solicitudes y permite saber en todo momento en que etapa del proceso se encuentra la solicitud.

Adicional a esto la herramienta permite conectarse con otras herramientas cliente para actualizar o descargar información requerida mediante el uso de web services u otras tecnologías.

El sistema trabaja en un arquitectura cliente servidor con lenguaje de programación php versión 5,2 o posteriores, motor de base de datos postgres 8,3 o posteriores y trabaja con un sistema operativo Linux (suse sles 10) con servidor apache, este es un software web tipo intranet.

6. EVALUACIÓN ECONÓMICA DEL PROYECTO.

6.1. Riesgo en fase de análisis

- Omisión de detalles en levantamiento de información
- Confusión de conceptos y redundancia de información
- Resistencia al cambio (los operadores creen que serán reemplazados por el software).
- Dimensionamiento errado del proyecto.

El riesgo en esta fase se cataloga como bajo.

6.2. Riesgo en fase de diseño

- Redundancia de información
- Ausencia de conocimiento en los procedimientos de la compañía
- Definición errónea de tiempos en fase de diseño
- Disminución de presupuestos por parte del cliente.
- Retrasos en levantamiento de información.
- Cambios de requerimientos que precisen cambios en el diseño.

El riesgo en esta fase se cataloga como medio.

6.3. Riesgo en fase de codificación

- Definición errónea de tiempos en fase de codificación
- Definición errónea de diseños
- Retrasos en fase de diseños.
- Cambios de requerimientos que precisen cambios en la codificación.
- Ausencia de herramientas de trabajo adecuadas (incluye instalaciones y equipos).

El riesgo en esta fase se cataloga como medio.

6.4. Riesgo en fase de pruebas

- Definición errónea de tiempos en fase de pruebas
- Definición errónea de diseños.
- Retrasos en fase de codificación.
- Cambios de requerimientos que precisen cambios en la aplicación.
- Ausencia de herramientas de trabajo adecuadas (incluye instalaciones y equipos).

El riesgo en esta fase se cataloga como medio.

6.5. Riesgo en fase de implementación

- Definición errónea en tiempo de fase de implementación.
- Ejecución errónea o incompleta de pruebas.
- Retrasos en fase de pruebas.
- Cambios de requerimientos que precisen cambios en la aplicación.
- Dimensionamiento errado de requisitos mínimos de instalación.

El riesgo en esta fase se cataloga como medio.

6.6. Riesgo en fase de mantenimiento

- Ejecución errada de scripts de backups
- Errores en capacitación y mal uso de la aplicación
- Perfilamiento errado de los usuarios.
- Dimensionamiento errado de requisitos mínimos de instalación.

El riesgo en esta fase se cataloga como medio.

7. PRESUPUESTO DETALLADO

7.1. Costo de infraestructura física

Tabla 3 Costo de la infraestructura física del proyecto

Infraestructura		
Concepto	Cant.	Valor Total
Sevidor Hp Proliant ML350, 24Gb Memoria, Procesador Intel® Xeon® E5-2400 v2 10 Nucleos	1	6000000
Disco duro HP de gama media, 3 TB, 3 G, SATA LFF (3,5 pulgadas), 7.200 rpm	4	1000000
Instalacion y configuracion server (Hora consultor)	14	700000
Total		7700000

7.2. Costo Total del Proyecto

Tabla 4 Tabla de costos generales del proyecto

Tabla 1 Tabla de Costos Generales del Proyecto		
Costo Ingeniero Junior		
Concepto	Costo	
Salario Mensual	3800000	
Salario Diario	126667	
Valor Hora	15833	
Personal		
Concepto	Horas	Costo
Recopilación De Información	48	\$ 760.000
Estudio y descripción de la situación inicial	48	\$ 760.000
Definición de la metodología de diseño e implantación	16	\$ 253.333
Levantamiento de Requerimientos	96	\$ 1.520.000
Fase de diseño	336	\$ 5.320.000
Fase de codificación	250	\$ 3.958.333
Fase de pruebas	100	\$ 1.583.333
Capacitación	80	\$ 1.266.667
Fase de Implementación	80	\$ 1.266.667
Total	1054	\$ 16.688.333

El costo total del proyecto es de \$ 24.388.333

8. BENEFICIOS DE IMPLEMENTACIÓN

8.1. Operacionales

A nivel operacional se listan los siguientes beneficios:

- Disminución de la labor operativa de las oficinas de servicio (este beneficio se reporta como consecuencia de permitir a las líneas de servicio asesorar telefónicamente a los afiliados)
- Descongestión y mayor velocidad de atención en las oficinas de servicio.
- Solicitar los documentos que realmente requiere el afiliado para su pensión lo que lleva a tener un mejor control de la documentación.
- Crecimiento de la capacidad de atención a afiliados.

8.2. De gestión

A nivel de gestión tenemos los siguientes beneficios:

- Tener un control real de la etapa en la que se encuentra un trámite, esto permite identificar cuellos de botella y atacarlos en tiempo real para hacer más efectiva la atención al afiliado que en su mayoría son personas de la tercera edad o son personas en situaciones de invalidez.
- Unificación de bases de datos, que permite realizar proyecciones más fácil y claramente.
- Mayor control de los Gerentes para proyectar capitales a provisionar ya que se conocen las fechas de vencimiento legal de las solicitudes.
- Reducción de las tasas de acciones legales contra la compañía, debido a que hay claridad en la información y etapa real en la que se encuentra.

8.3. Estratégicos

A nivel estratégico los beneficios son:

- Permitir a la compañía realizar proyecciones reales, basadas en datos unificados en un solo aplicativo.
- Reducir costos de atención a afiliados con la implementación de call center.
- Mejorar la imagen respecto a la atención a sus afiliados, lo que llevará a tener una mejor “publicidad” de cara a nuevos ingresos de afiliados.
- Permitir crecer en productos a partir de un sistema montado.

8.4. De Infraestructura

Los beneficios de infraestructura son:

- Con la tercerización de funciones se aprovechan mejor las infraestructuras físicas ya que se dispone de más espacio.
- Reducción de costos de administración por reducción en personal.

8.5. De Tecnologías de la Información (IT)

A nivel de tecnologías de la información podemos listar los siguientes beneficios:

- Mayor facilidad de controlar la información de las solicitudes de pensión ya que se encuentran consolidadas en una sola base de datos.
- Optimización en tiempos de respuesta del servidor ya que las consultas se ejecutan solo a un sistema.
- Mayor facilidad en la ejecución de tareas Bash y backups.

9. ALCANCES DEL PROYECTO

El proyecto de Servicios Previsionales se desarrolla con el mínimo viable de ejecución bajo una metodología Scrum, el alcance del proyecto compromete a todas las áreas de la compañía en las principales de Bogotá y Medellín, así como tiene alcance a nivel nacional para consulta y creación de solicitudes en todas las oficinas de servicio disponibles.

Este proyecto se desarrolla de manera modular, lo que permitirá ir creciendo en fases que permitirán trabajar sobre la herramienta a medida que esta crece.

10.LIMITACIONES DEL PROYECTO

El proyecto encuentra limitaciones en cuanto a que no se profundizó en la distribución de cargas de trabajo a nivel de servidores, por otra parte debe existir un componente de conexión con el sistema padre del cliente para evitar pérdida o redundancia en los datos y trabajos operativos de replicación.

11. CRONOGRAMA DE ACTIVIDADES

A continuación se lista el cronograma de actividades, sus responsables: y las fechas en las que se inició y entregaron las respectivas actividades.

Tabla 5 Cronograma de Trabajo

Fase	Tarea	Responsable	Fecha Inicio	Fecha Entrega
Fase de Análisis	Entrega de Histórico De Solicitudes para Volumetría	Protección	03/08/2015	10/08/2015
Fase de Análisis	Listar Procesos Afectados	Protección	03/08/2015	10/08/2015
Fase de Diseño	Levantamiento de Requerimientos	Imagine	13/08/2015	31/08/2015
Fase de Diseño	Diseño de Pantallas	Imagine	01/09/2015	19/09/2015
Fase de Diseño	Diseño de Base de Datos	Imagine	01/09/2015	19/09/2015
Fase de Codificación	Codificación del Sistema	Imagine	21/09/2015	19/10/2015
Fase de Pruebas	Pruebas de Aplicación	Protección	20/10/2015	04/11/2015
Fase de Implementación	Montaje del Sistema	Imagine	04/11/2015	18/11/2015
Fase de Implementación	Capacitación Áreas	Protección	04/11/2015	18/11/2015
Fase de Implementación	Salida a Producción	Imagine	01/12/2015	01/12/2015

En total el proyecto se tardó 3 meses en su diseño e implementación.

12.RECOMENDACIONES

El proyecto descrito en este trabajo se pensó como una herramienta con un mínimo viable para operar, se recomienda hacer una fase 2 que amplíe las diferentes etapas del flujo de trabajo y permita tener aún más control de la operación.

Se recomienda prestar atención a los reportes y realizar una separación del reporte general dentro de cada área involucrada, ya que el crecimiento del mismo es exponencial y la generación de dicho reporte puede afectar el rendimiento de la aplicación.

Otro punto importante a tener en cuenta es hacer una revisión anual del comportamiento del servidor teniendo en cuenta que se incrementaran las acciones en dicho servidor al ser la aplicación de uso nacional.

13. CONCLUSIONES

- Los sistemas tipo workflow permiten agilizar y controlar el funcionamiento de las tareas diarias de una compañía.
- Mediante el uso de metodologías ágiles se reducen tiempos de desarrollo y se aumenta la satisfacción del usuario al sentirse involucrado con el desarrollo.
- El desarrollo web permite facilitar el acceso a la aplicación debido a que todas las máquinas hoy en día disponen de un navegador web, lo que permite reducir tiempos de instalación y de mantenimiento, ya que el mismo se realiza del lado del servidor.
- Es importante tener herramientas que provean de información a los administradores del proceso, ya que estas facilitarán la toma de decisiones claves en cualquier compañía.
- El correcto modelamiento de un sistema de información permite realizar un ciclo de vida del sistema más confiable y óptimo, esto reducirá los tiempos en las fases de diseño, codificación, implementación y mantenimiento.

BIBLIOGRAFÍA

- **Borja Obergozo Arana**, CURSO PRÁCTICO AVANZADO DE POSTGRESQL: LA BASE DE DATOS MÁS POTENTE, Altaria, 2015.
- **Cantone Dante**, ADMINISTRACION DE STORAGE Y BACKUPS, RAMA, 2012.
- **Cibelli Christian**, PHP PROGRAMACION WEB AVANZADA, Alfaomega, 2012.
- **Elmasri Ramez/ Navathe Shamkant**, FUNDAMENTOS DE SISTEMAS DE BASES DE DATOS - 5 Edición, PRENTICE HALL, 2014.
- **Gomez Julio**, LINUX ADMINISTRACION DE SISTEMAS GNU LINUX, RAMA, 2011.
- **Jesus Gonzalez Lorca**, SISTEMAS WORKFLOW FUNCIONAMIENTO Y METODOLOGIA DE IMPLANTACION, Universidad De Alcala De Henares, 2014.
- **Luis F. Diaz Dominguez**, SISTEMAS DE GESTION INTEGRADA PARA LAS EMPRESAS (ERP), Universidad De Alcala De Henares, 2014.
- **Normatividad**, CARTILLA DE SEGURIDAD SOCIAL Y PENSIONES, Legis, 2016.
- **Piattini / Garcia / Garcia Ignacio / Pino**, CALIDAD DE SISTEMAS DE INFORMACION 2 EDICION, RAMA, 2014.

WEBGRAFÍA

- Gestión de procesos de Negocio, <http://www.pixelware.com/workflow-flujo-trabajo.htm>
- Gestión en el tercer milenio:
http://sisbib.unmsm.edu.pe/bibvirtual/publicaciones/administracion/v03_n6/tecnologias.htm
- ¿Qué es un workflow o flujo de trabajo?, <http://www.gestion.org/economia-empresa/gestion-administrativa/29867/que-es-workflow-o-flujo-de-trabajo/>
- Preguntas Frecuentes,
<https://www.proteccion.com/wps/portal/proteccion/web/menu-global/preguntas-frecuentes>
- PostgreSQL Documentación 8.3
<http://www.postgresql.org/docs/8.3/static/index.html>
- Documentación php <http://php.net/docs.php>
- Suse Linux <https://www.suse.com/>

ANEXOS DEL DOCUMENTO

- Presentación en Power Point realizada para la sustentación del proyecto
- Modelo entidad Relación.
- Documento en Word con manual de la Aplicación.